

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

Dpto. de INGENIERÍA MECÁNICA



INGENIERÍA INDUSTRIAL

PROYECTO FIN DE CARRERA

**DESARROLLO DE UNA APLICACIÓN
PARA LA DOCENCIA DEL SISTEMA
DIÉDRICO**

AUTOR: ISABEL ROA PRIETO

TUTOR: CAROLINA ALVAREZ CALDAS

Índice de Contenido

TÍTULO I: INTRODUCCIÓN	1
1. Introducción.....	2
2. Objetivos	4
3. Nuevas Tecnologías en docencia	5
3.1. Nuevas Tecnologías para el Dibujo Técnico	6
TÍTULO II: ASPECTOS TEÓRICOS	9
4. El Sistema Diédrico.....	10
4.1. Elementos del Sistema	10
4.2. El Punto	12
4.2.1. Representación del Punto	13
4.2.2. Posiciones Particulares del Plano	13
4.3. Recta.....	16
4.3.1. Representación de la recta.....	16
4.3.2. Ecuaciones de la recta	19
4.3.3. Posiciones particulares de la recta	20
4.3.4. Posiciones relativas de dos rectas.....	26
4.4. Plano.....	27
4.4.1. Representación del plano.....	27
4.4.2. Ecuaciones del plano	29
4.4.3. Posiciones particulares del plano	30
5. MATLAB.....	40
5.1. Introducción al MATLAB.....	40
5.1.1. Desarrollo de algoritmos y aplicaciones.....	40
5.2. Interfaces Gráficas con MATLAB	41
5.2.1. Estructura de los gráficos de MATLAB	42
5.2.2. Creación de controles gráficos: Comando uicontrol	44
5.3. Guide	46
TÍTULO III: SISTEMA DIÉDRICO CON MATLAB	48
6. Elecciones Previas	49
6.1. Funcionalidad	49
6.2. Interfaz Gráfica.....	49

6.3.	Geometría	49
6.4.	Nomenclatura y elementos.....	51
7.	Diseño Técnico	53
7.1.	Diseño de Interfaz Gráfica	53
7.2.	Interacción con el Usuario.....	58
7.3.	Implementación	60
7.3.1.	Inicializar Espacio de Trabajo: Función LIMPIAR.....	60
7.3.2.	Creación de Elementos: Función NUEVO	62
7.3.3.	Borrado de Elementos: Función BORRAR	70
7.3.4.	Dibujar Elementos: Función DIBUJAR	76
7.3.5.	Funciones Auxiliares Generales.....	99
TÍTULO IV: CONCLUSIONES Y DESARROLLOS FUTUROS.....		104
8.	Conclusiones	105
9.	Desarrollos Futuros.....	106
BIBLIOGRAFÍA		107
ANEXOS		108
Manual de Usuario de la Aplicación.....		109

Índice de Figuras

Fig. 4.1 Espacio en el Sistema Diédrico.....	10
Fig. 4.2 Octantes en el Sistema Diédrico.....	11
Fig. 4.3 Representación del Sistema Diédrico.....	12
Fig. 4.4 Ejes de Coordenadas en los Planos de Proyección	12
Fig. 4.5 Representación del Punto	13
Fig. 4.6 Alfabeto del Punto	14
Fig. 4.7 Representación de la Recta.....	17
Fig. 4.8 Representación de la Recta a partir de dos Puntos.....	18
Fig. 4.9 Recta Inclínada u Oblicua.....	20
Fig. 4.10 Recta Horizontal	21
Fig. 4.11 Recta Frontal.....	22
Fig. 4.12 Recta de Punta.....	22
Fig. 4.13 Recta Vertical.....	23
Fig. 4.14 Recta Paralela a LT.....	24
Fig. 4.15 Recta que Corta a LT.....	24
Fig. 4.16 Recta de Perfil.....	25
Fig. 4.17 Representación del Plano.....	28
Fig. 4.18 Representación del Plano a partir de un Punto y una Recta	29
Fig. 4.19 Plano Oblicuo.....	31
Fig. 4.20 Plano Horizontal	32
Fig. 4.21 Plano Frontal.....	33
Fig. 4.22 Plano de Perfil.....	34
Fig. 4.23 Plano Vertical.....	35
Fig. 4.24 Plano de Canto.....	36
Fig. 4.25 Plano Paralelo a LT	37
Fig. 4.26 Plano que pasa por LT.....	38
Fig. 5.1 Jerarquía Gráfica de MATLAB	42
Fig. 5.2 Operación Software del GUI	47
Fig. 6.1 Nomenclatura del Punto	51
Fig. 6.2 Nomenclatura de la Recta	52
Fig. 6.3 Nomenclatura del Plano	53
Fig. 7.1 Inicialización del Espacio de Trabajo.....	54
Fig. 7.2 Zona de Actualización	55
Fig. 7.3 Zona de Memoria.....	55
Fig. 7.4 Zona de Dibujo	56
Fig. 7.5 Zona de Pie.....	56
Fig. 7.6 Interacción con el Usuario.....	59
Fig. 7.7 Diagrama de Limpiar	61
Fig. 7.8 Espacio de Trabajo Inicial.....	62
Fig. 7.9 Diagrama de Nuevo	63
Fig. 7.10 Diagrama de Nuevo: Sacar atributos Puntos.....	65
Fig. 7.11 Aviso: Máximo de Puntos Superado	66
Fig. 7.12 Aviso: Punto Repetido.....	66
Fig. 7.13 Aviso: Coordenadas no Numéricas	66
Fig. 7.14 Aviso: Número de Coordenada Incorrecto.....	66
Fig. 7.15 Diagrama de Nuevo: Sacar atributos Rectas	67

Fig. 7.16 Aviso: Máximo de Rectas Superadas	68
Fig. 7.17 Aviso: Recta Repetido	68
Fig. 7.18 Aviso: Nº de Puntos para Recta Incorrecto	68
Fig. 7.19 Aviso: Puntos de la Recta Iguales.....	68
Fig. 7.20 Diagrama de Nuevo: Sacar atributos Planos	69
Fig. 7.21 Aviso: Máximo de Planos Superados	70
Fig. 7.22 Aviso: Plano Repetido	70
Fig. 7.23 Aviso: Puntos para Crear Planos Repetidos	70
Fig. 7.24 Aviso: Puntos Forman la Misma Recta.....	70
Fig. 7.25 Aviso: Punto Dentro de la Recta	70
Fig. 7.26 Aviso: Las Rectas no forman un Plano	70
Fig. 7.27 Diagrama de Borrar	72
Fig. 7.28 Aviso: Varios Elementos Seleccionados.....	73
Fig. 7.29 Aviso: Elemento utilizado.....	73
Fig. 7.30 Diagrama Función borrarAuxiliar	75
Fig. 7.31 Diagrama de Dibujar.....	77
Fig. 7.32 Diagrama de Dibujar: Preparar Elementos	79
Fig. 7.33 Reajuste de Escalas.....	82
Fig. 7.34 Ejemplo: Puntos 2D.....	84
Fig. 7.35 Ejemplo: Puntos 3D.....	85
Fig. 7.36 Diagrama de Dibujar: Rectas en 2D.....	87
Fig. 7.37 Ejemplo: Rectar Vertical y de Punta 2D	88
Fig. 7.38 Ejemplo: Rectas de Perfil 2D	88
Fig. 7.39 Ejemplo; Recta Paralela a LT 2D	89
Fig. 7.40 Ejemplo: Rectas 2D	89
Fig. 7.41 Ejemplo: Rectas 3D	91
Fig. 7.42 Diagrama Función lineaContinua.....	93
Fig. 7.43 Ejemplo: Planos 2D	95
Fig. 7.44 Dibujo de Planos en 3D	96
Fig. 7.45 Dibujo de Planos en 3D	97
Fig. 7.46 Trazas en el plano Horizontal.....	98
Fig. 7.47 Trazas en el plano Vertical.....	99
Fig. 7.48 Diagrama de sacaIndices	101
Fig. 7.49 Diagrama de relacionRectas.....	103

Índice de Tablas

<i>Tabla 4.1 Visibilidad de las Rectas según sus Trazas</i>	17
<i>Tabla 4.2 Rectas vs Vectores Directores</i>	26
<i>Tabla 4.3 Planos vs Ecuaciones</i>	39
<i>Tabla 7.1 Recta vs elemento oculto "rectas"</i>	86
<i>Tabla 7.2 Planos vs elemento oculto "planos"</i>	94

TÍTULO I: INTRODUCCIÓN

1.Introducción

Por lo general, las asignaturas de Dibujo Técnico presentan gran dificultad para los alumnos, ya que requieren una visión espacial que no todos tienen, además del alto grado de abstracción necesario para comprender ciertos procesos. Por ello, en el dibujo técnico es recomendable usar herramientas multimedia interactivas que permitan a los alumnos, además de tener un aprendizaje activo, lograr una visión espacial más sencilla y clara que la que se obtiene sobre el papel.

Ante esta dificultad palpable se ha decidido realizar una herramienta que permita una visualización rápida de los elementos, para que el propio usuario introduzca elementos con un sencillo manejo y una programación clara, ya que, aunque esta aplicación en estos momentos se centra en la representación de los elementos geométricos simples (punto, recta y plano), se pretende un programa más completo donde se puedan representar intersecciones entre elementos, elementos más complejos...

El MATLAB es un software muy potente utilizado en ingenierías, por lo que es conocido por gran parte de los alumnos, lo que hace que sea mucho más familiar para ellos, y además facilita una futura ampliación.

El MATLAB soporta todo el proceso de análisis de datos, desde la adquisición de datos de dispositivos externos y bases de datos, pasando por el preprocesado, visualización y análisis numérico, hasta la producción de resultados con calidad de presentación y, además proporciona herramientas interactivas y funciones de línea de comandos para operaciones de análisis de datos. Es uno de los softwares matemáticos más potentes del mercado, que se basa en la utilización de matrices.

El programa de esta aplicación pretende la representación y simulación de puntos rectas y planos a través de coordenadas de uno o varios puntos. El trabajo principal de este proyecto es todo el conjunto de transformaciones geométricas que engloba, que a través del MATLAB se hace mucha más sencillo, rápido y fiable. El programa pretende, a través de un grupo de coordenadas que se procesan con un conjunto de herramientas, ecuaciones y validaciones matemáticas, devolver los puntos, rectas y planos ya calculados, con una gran rapidez (casi instantáneo a la vista).

Aunque la herramienta gráfica utilizada de MATLAB no sea una de las más potentes, facilita un gran número de librerías y funciones que ayudan a la visualización, y permiten la comprensión de las imágenes.

Se ha elegido MATLAB como software para este trabajo por la gran rapidez, facilidad y potencia para el tratamiento con matrices y vectores, muy utilizados para la geometría, además de la multitud de posibilidades para las interfaces gráficas.

Estructura del proyecto

El proyecto consta de cuatro títulos generales, que engloban partes muy diferentes entre sí:

TITULO I: Introduce al lector en la necesidad de una aplicación de estas características, estableciendo los objetivos que se pretenden alcanzar. Se divide en tres capítulos:

- Capítulo 1, es una pequeña introducción al proyecto, donde se pone de manifiesto el por qué del software elegido.
- Capítulo 2, presenta los objetivos a superar frente a este proyecto.
- Capítulo 3, hace hincapié en la necesidad de adaptar la enseñanza a las Nuevas Tecnologías y aprovechar las grandes ventajas que se brindan muy especialmente en Dibujo Técnico.

TITULO II: Se centra en los conceptos básicos que se han necesitado para el buen desarrollo del proyecto.

- Capítulo 4, para la comprensión del trabajo es necesario fundamentos básicos del Sistema Diédrico, como es la representación de los elementos básicos y la necesidad de relacionarlos con las formulas geométricas fundamentales para el conseguir un dibujo claro y simple. Este punto es de gran importancia, ya que estos conceptos son los que esta aplicación pretende ayudar a comprender.
- Capítulo 5, para centrarse más en la forma de codificación, se hace una pequeña introducción al MATLAB y más detenidamente a la interfaz gráfica utilizada, con especial énfasis en los elementos utilizados con más dificultad técnica.

TÍTULO III: Explica detalladamente la forma de llevar a cabo la aplicación.

- Capítulo 6, antes de iniciar cualquier proyecto, lo primero es tomar una serie de elecciones sobre cómo hacerlo y cuál es el resultado final esperado.
- Capítulo 7, hace una descripción funcional detallada, explicando la secuencia del programa con la ayuda de diagramas y figuras en caso de ser necesarias.

TÍTULO IV: Resultados y expectativas.

- Capítulo 8, una vez finalizado el proyecto se evalúa si se han cumplido los objetivos propuestos.
- Capítulo 9, propone una serie de ampliaciones a la aplicación ayudar, para seguir mejorando el objetivo principal de esta.

Entre los anexos de este documento se ha incluido un manual de usuario detallado, como documento aparte.

IMPORTANTE: Está aplicación está preparada para trabajar con una configuración de pantalla de 1280 por 1024 pixeles.

2.Objetivos

Esta aplicación nace de la necesidad de disponer de una herramienta interactiva para que el alumno pueda comprender los conceptos básicos de esta asignatura, por lo que debe ser una herramienta que permita una visibilidad clara de los elementos dibujados, con posibilidades de que el alumno maniobre con objeto de aclarar sus dudas.

Se pretende que el usuario sea independiente y, además que la herramienta sea sencilla de utilizar para que no requiera explicación del profesor, o en caso de ser necesaria, baste una explicación mínima; por otra parte, pretende ser dinámico porque si es aburrido desmotivará al alumno.

También se pretende que este programa sea ampliado por otra persona, que en principio es ajena a él, por lo que el código es fácil de entender, limpio y bien estructurado, con gran cantidad de comentarios, que permiten, en un rápido vistazo, localizar las órdenes a modificar para mejorarlo o ampliarlo.

Además, se utilizan las ventajas que proporciona MATLAB como software de cálculo orientado a matrices.

3.Nuevas Tecnologías en docencia

La Enseñanza hoy más que nunca necesita renovarse, dando respuesta a las variadas demandas sociales y laborales. Por eso, incluir aplicaciones informáticas en el ámbito escolar/universitario constituye una acción necesaria y urgente, al servicio de una educación transformadora y beneficiosa para aprender con mayor rapidez y facilidad, brindando al alumno la posibilidad de investigar, adaptándose a la tecnología actual y a los cambios constantes. Es obligación del docente desarrollar y apoyar mejores ambientes de aprendizaje, a través de los contenidos, generando propuestas tecnológicas, asesorando cuando se requiere apoyo. Este es el rol del nuevo docente, el de facilitador o mediador del aprendizaje.

Cada vez son más los centros educativos que adquieren, en ocasiones con gran esfuerzo, equipos informáticos multimedia, confiando en sus posibilidades educativas. Lamentablemente en muchos casos dichos equipos apenas se utilizan para mejorar el proceso de aprendizaje de los alumnos por no estar los profesores preparados para ello.

Cuando se tenga que evaluar y ver las ventajas de un programa multimedia, habrá que hacer siempre referencia a si utiliza estas ventajas que la Multimedia aporta. No se puede hacer lo mismo que antes, pero ahora usando el ordenador: o el ordenador aporta algo, o si no es preferible seguir haciendo las cosas como antes. Sin embargo, el uso del ordenador abre multitud de posibilidades nuevas:

1. **Más atractivo:** el uso de los ordenadores es más atractivo para los alumnos. Las Nuevas Tecnologías despiertan interés en los estudiantes jóvenes. Pero es también más atractivo porque se incluyen colores, sonidos, vídeos... Hay programas que son como una aventura: los chicos creen que están jugando, pero están aprendiendo. Los estudiantes no están pasivos sino que tienen que interactuar continuamente y son los protagonistas en su proceso de aprendizaje.
2. **Interactividad:** Desde hace muchos años el papel de los estudiantes ha sido pasivo: el profesor explicaba y los alumnos escuchaban, tomaban apuntes, y de vez en cuando una o dos veces por clase- podían 'interactuar' con el profesor. La Multimedia permite que los alumnos estén continuamente interactuando con el ordenador, siendo los protagonistas de su aprendizaje. El profesor es el que prepara esos materiales, y ayuda y dirige a los alumnos individualmente.
3. **Continuo feed-back:** que permiten estos programas. En una clase convencional, el profesor apenas tiene tiempo para resolver todas las dudas de los alumnos, o no tiene tiempo para evaluarles, para comprobar que han ido asimilando los conocimientos que se han transmitido durante la clase. En muchas ocasiones los mismos alumnos no se atreven a formular preguntas al profesor, por vergüenza, o por miedo a hacer el ridículo. Con una aplicación Multimedia los alumnos pueden formular sus preguntas al ordenador cuantas veces quieran, y el ordenador contesta siempre, sin cansarse y sin enfadarse. Nunca un alumno dejará de consultar un ordenador por vergüenza. Por otro lado, el profesor puede utilizar el ordenador para evaluar continuamente el aprendizaje de los alumnos.

4. **Amoldable al aprendizaje de cada alumno:** Un mismo programa puede permitir distintos niveles de dificultad, o facilitar que los alumnos más aventajados puedan ir más deprisa que los alumnos con dificultades. Esto sí es una gran ventaja frente a la enseñanza tradicional; hoy en día se ha conseguido que casi el 100% de los jóvenes de nuestros países realicen estudios hasta los 16 años, lo cual es un avance, pero supone una dificultad no pequeña para los profesores: tienen que dar clases y enseñar a alumnos con capacidades distintas. Si no se resuelven estas limitaciones, el profesor acabará impartiendo una enseñanza en un nivel adaptado a la media intelectual de sus alumnos, de modo que los más aventajados no aprenden y se aburren, y los alumnos con dificultades caen en la desmotivación al no asimilar las enseñanzas del profesor por tener un ritmo de aprendizaje más lento que la media de los estudiantes de su clase.
5. **Posibilidad del control directo del trabajo realizado por cada alumno:** Aunque cada alumno esté trabajando solo con el ordenador, y en un nivel de dificultad distinto cada uno, hay programas que van registrando el nivel alcanzado por el usuario e incluso especifican aquellos aspectos en que suele fallar más frecuentemente el alumno. De esta manera el profesor tiene elementos de juicio para evaluar el trabajo de cada alumno con más objetividad, y evita que los alumnos se tomen las clases en las que se utilizan programas multimedia como un juego.
6. **Acceso a una información más real:** los libros incluyen información escrita y algunas fotografías (pocas), pero no pueden incluir sonidos, vídeos, etc.

Esta característica es especialmente eficaz en la enseñanza de idiomas: practicar un idioma extranjero con un compañero de la misma clase no es un contexto real, porque ninguno de los dos suele hablar en ese idioma, sin embargo, intercambiar correos electrónicos con estudiantes de otro país usando un idioma como en sí es un contexto real.

No todos los programas multimedia tienen estas características: por lo tanto, la primera habilidad que tiene que tener un profesor que utilice aplicaciones multimedia tiene que ser precisamente ésta: distinguir aquellas que realmente aporten claras ventajas en el proceso de aprendizaje de los estudiantes.

3.1. Nuevas Tecnologías para el Dibujo Técnico

El Dibujo Técnico tiene una trascendental importancia en estudios universitarios de carreras técnicas como Ingeniería y Arquitectura, ya que para que el desarrollo de su vida laboral éstos profesionales requerirán, en muchos de los casos, el dominio de éste lenguaje. Por esta razón el alumno debe estar bien preparado en estas materias y se deben poner todos los medios posibles para una enseñanza óptima de las asignaturas de Expresión Gráfica.

Sin embargo, el material didáctico empleado en las escuelas técnicas para el estudio de los Sistemas de Representación, como parte de la Geometría Descriptiva, presenta en general dificultades de aprendizaje para los estudiantes porque el material el estudio en esta materia es, fundamentalmente, escrito, lo cual supone una dificultad en la comprensión para un número importante de alumnos porque se precisa un “alto grado de abstracción necesario

para comprender ciertos procesos, lo que tiene que facilitarse con métodos perceptivos y reflexivos”[1].

Por norma general la proporción de suspensos en relación con los estudiantes matriculados es mayor en las asignaturas de dibujo técnico que en otras materias. Se señalan como causas:

1.- La dificultad intrínseca de del dibujo técnico, que requiere aptitudes y conocimiento del Sistema Diédrico.

2.- La forma en que se imparten estas asignaturas, que tienen un temario amplio y de difícil explicación a pesar del esfuerzo que realiza el profesor, porque “si para el profesor resulta difícil compaginar su explicación con la construcción en la pizarra del ejercicio, mucho más difícil resulta para el alumno atender a la explicación, copiar el ejercicio en sus apuntes y además intentar entenderlo, llegando a la situación de estar copiando y no saber lo que se copia. El resultado es que el alumno tiene en sus apuntes una construcción sin ninguna explicación escrita y sin saber de dónde sale cada línea, cuál ha sido la primera y cuál la última. Así, cuando el alumno se pone a estudiar en casa no puede entender el dibujo realizado si no va parte por parte, tiene que repetir la construcción paso a paso y eso se convierte en algo muy difícil si no tiene apuntada la secuencia”[3].

En la bibliografía disponible los ejercicios vienen explicados con un desarrollo escrito paso a paso, se explica la resolución de los ejercicios de manera detallada, pero el problema se encuentra en que la figura que acompaña a la explicación es la construcción final del problema, por lo que el alumno ha de hacer un ejercicio de imaginación además de entendimiento para comprender la construcción.

Todos estos problemas llevan al alumno que quiera superar la asignatura al empleo de mucho tiempo para poder entenderla, lo que puede suponer un abandono de otras asignaturas, además, el alumno suele buscar ayudas fuera de la universidad lo que le va a ocasionar un gasto económico considerable” [2].

Se trata de superar estas dificultades a través de métodos de aprendizaje que permitan un diálogo constante entre teoría y experimentación, para, no solo comprender los conceptos, sino también aplicarlos mediante el dominio de los recursos materiales, ya sean tradicionales o modernos en soporte informático.

Facilitar el aprendizaje de los alumnos es el objetivo de este trabajo. Se diseña un programa informático que permite al alumno “interactuar, situar los elementos o a ellos mismos, en diferentes puntos del espacio para comprobar in situ lo que es difícil de imaginar” [3].

Las páginas estáticas no sirven para acercar el mundo de la Geometría Descriptiva a los alumnos.

Se pretende superar estas dificultades a través de un diálogo constante entre teoría y experimentación, que permita no sólo comprender los conceptos, sino también aplicarlos mediante el dominio de los recursos materiales, ya sean tradicionales o modernos en soporte informático.

El soporte informático en el dibujo técnico ayuda a desarrollar la visión espacial y la capacidad de abstracción. Permite al alumno marcar su propio ritmo y favorece la retroalimentación del proceso de aprendizaje.

Para el uso de este programa se ha redactado un sencillo manual de fácil comprensión por los estudiantes porque la pretensión de este programa es hacer atractivo el estudio de las asignaturas de Dibujo Técnico y estimular a los alumnos a formular nuevas hipótesis y abrir nuevos horizontes, que contribuyan a familiarizarse con el uso del Sistema Diédrico.

TÍTULO II: ASPECTOS TEÓRICOS

4.El Sistema Diédrico

El Sistema Diédrico ó de G. Monge es un sistema de representación geométrico de los elementos del espacio sobre un plano, es decir, la reducción de las tres dimensiones del espacio a las dos dimensiones del plano, utilizando una proyección ortogonal sobre dos planos que se cortan perpendicularmente. Para generar las vistas diédricas, uno de los planos se abate sobre el segundo.

El Sistema Diédrico de representación surge por la necesidad de representar elementos tridimensionales en el papel, formato de dos dimensiones.

4.1. Elementos del Sistema

Como muestra en la Fig. 4.1 en el Sistema Diédrico el espacio queda dividido en cuatro partes iguales, por medio de dos planos perpendiculares entre sí, llamados planos coordenados o de proyección, serán el Plano de proyección Vertical (PV) y Plano de Proyección Horizontal (PH). Estos dos, como cualquier par de planos que no presenten la particularidad de ser paralelos entre sí, se cortarán en una recta, recta conocida por Línea de Tierra (LT).

De modo que el espacio debido a estos dos planos queda dividido en cuatro partes iguales, cada una de las cuales recibe el nombre de diedro o cuadrante, cada uno de estos cuadrantes estarán enumerados como muestra en la Fig. 4.1.

El observador se supone situado en el punto del infinito anterior al primer cuadrante, por lo que la parte visible será lo dibujado en el primer cuadrante.

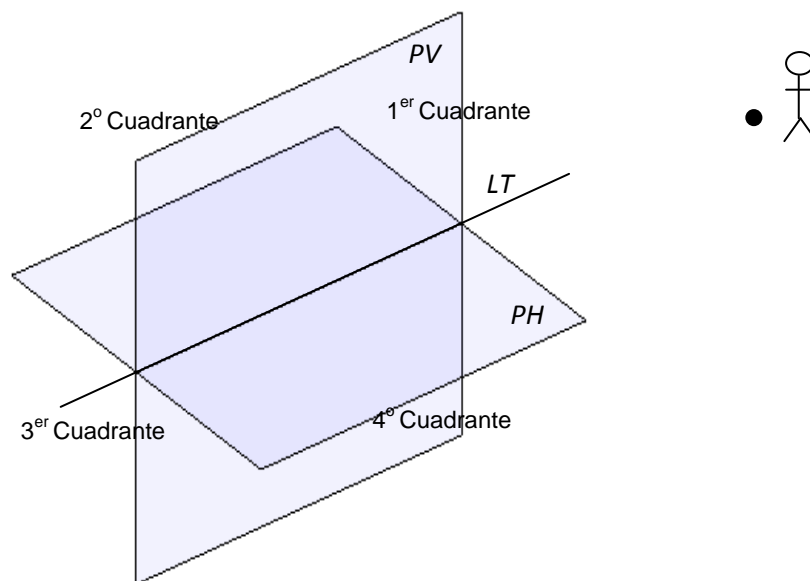


Fig. 4.1 Espacio en el Sistema Diédrico

Además de estos dos planos existen otros dos, que dividen los diedros mencionados en dos partes iguales. Estos planos forman 45° con los planos de proyección y se cortan entre ellos y a los planos de proyección en la Línea de Tierra. De este modo nuestro sistema queda dividido en ocho partes iguales a las que llamaremos octantes como muestra la Fig. 4.2, y a los dos nuevos planos causantes de esta segunda división planos bisectores.

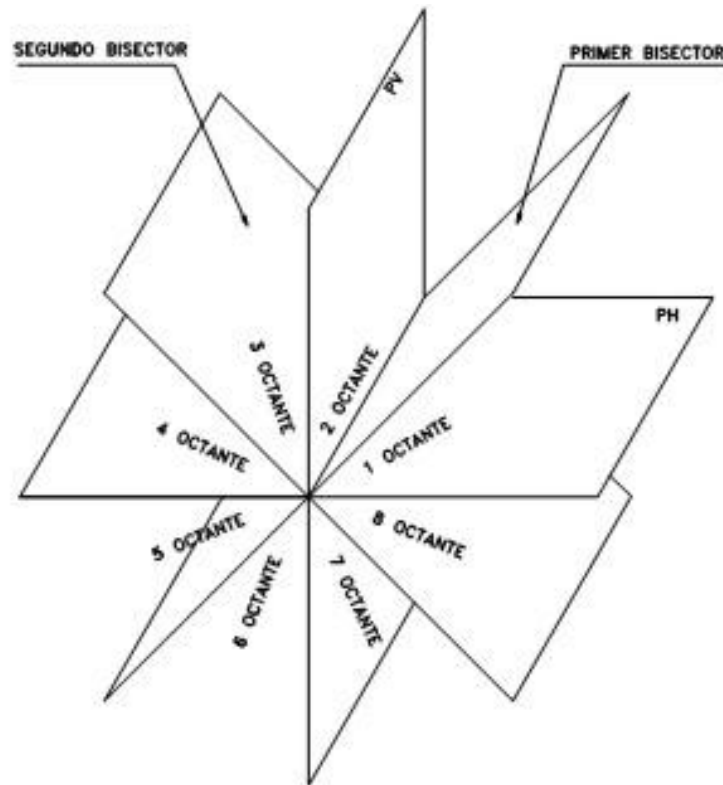


Fig. 4.2 Octantes en el Sistema Diédrico

Lo expuesto hasta el momento da una visión del sistema de representación en el espacio. Para representarlo en un plano habrá que abatir el Plano de proyección Horizontal sobre el Plano de proyección Vertical utilizando como eje de giro la propia Línea de Tierra como se muestra en la Fig. 4.3. De este modo, quedará como único elemento de referencia la Línea de Tierra. Así, pueden representarse las 3 dimensiones del espacio, en un plano de 2 dimensiones.

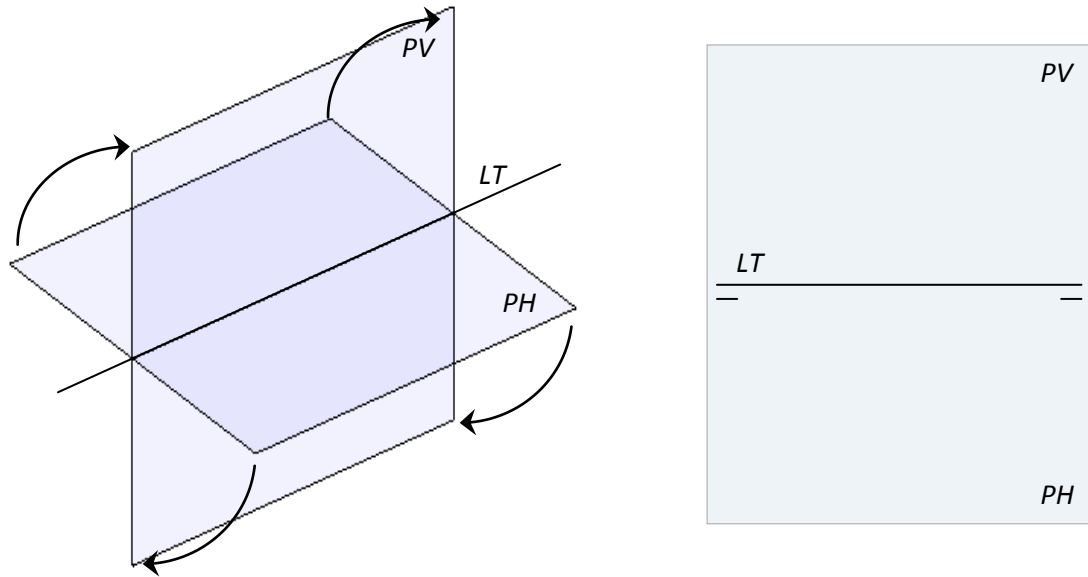


Fig. 4.3 Representación del Sistema Diédrico

En ocasiones, es necesario realizar una tercera vista o proyección del elemento que se está representando para su total definición y comprensión. Esta proyección se realiza sobre un tercer plano de proyección denominado Plano de Perfil.

Para la representación de los elementos que se explican a continuación, utilizando conceptos geométricos, se imaginarán un sistema de coordenadas X, Y, Z situados en los Planos de Proyección, tal como se muestra en la Fig. 4.4, de tal forma que se podrá crear elementos a través de coordenadas:

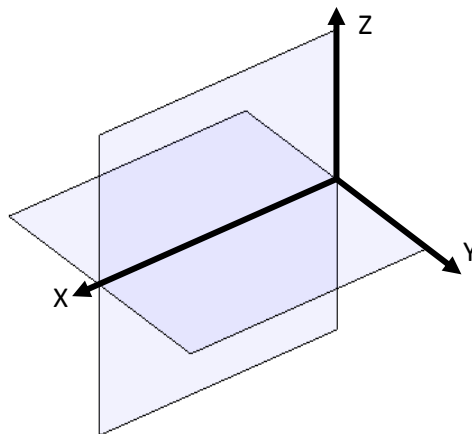


Fig. 4.4 Ejes de Coordenadas en los Planos de Proyección

4.2. El Punto

El punto es el elemento geométrico más simple. No tiene dimensiones, es inmaterial. Sólo tiene posición. Dos puntos definen una línea recta (un segmento). Tres puntos no alineados forman un plano (un triángulo).

4.2.1. Representación del Punto

En el Sistema Diédrico los puntos se representan con sus proyecciones en cada uno de los planos de proyección. Por ejemplo el punto P representado en la Fig. 4.5 se representa con la proyección vertical (P''), en el Plano de proyección Vertical, y la proyección horizontal (P'), en el Plano de proyección Horizontal.

La distancia que hay desde el punto al Plano Horizontal se denomina Cota y la distancia que hay desde el punto al Plano Vertical se denomina Alejamiento.

Como se muestra en la Fig. 4.5 la representación en el Sistema Diédrico de un punto cualquiera se hace a partir de una línea perpendicular a la Línea de Tierra, midiendo para la proyección vertical la cota del punto y para la proyección horizontal el alejamiento del punto.

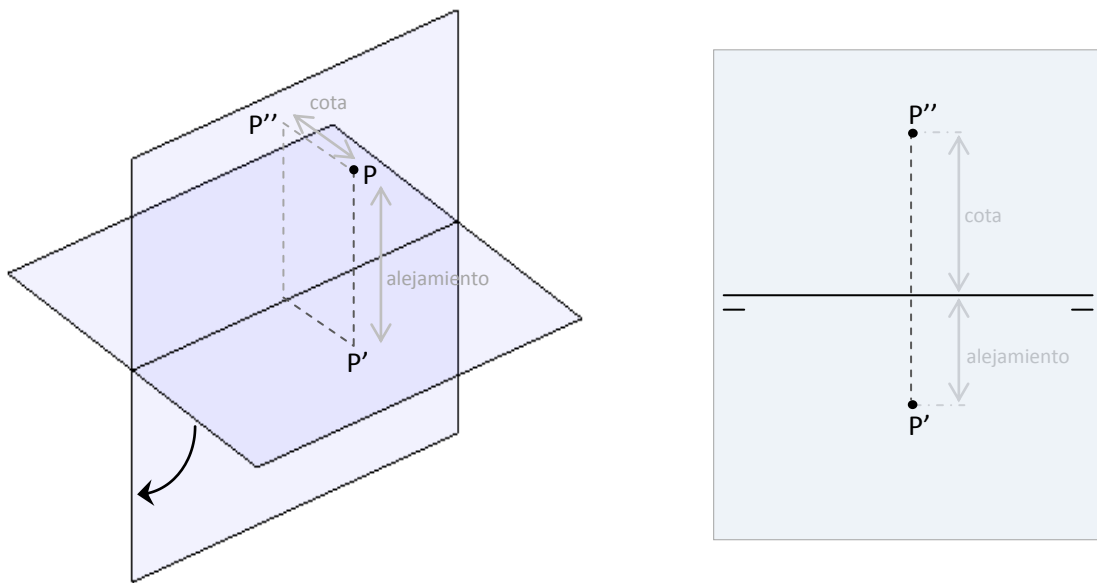


Fig. 4.5 Representación del Punto

Utilizando el sistema de coordenadas X, Y, Z situados en los planos de proyección, puede definirse cualquier punto dando sus coordenadas en la forma (x, y, z) , que se correspondería con:

- X= Desviación
- Y = Alejamiento
- Z = Cota

4.2.2. Posiciones Particulares del Plano

Un punto en el espacio puede tener cualquier situación, pero según sea su posición relativa a los planos de proyección y de los planos bisectores, presenta características de representación especiales.

En la Fig. 4.6 se muestran las 17 posiciones del punto a considerar:

- 4 en los planos de proyección

- 4 en los planos bisectores
- 8 en cada uno de los octantes
- 1 en la LT

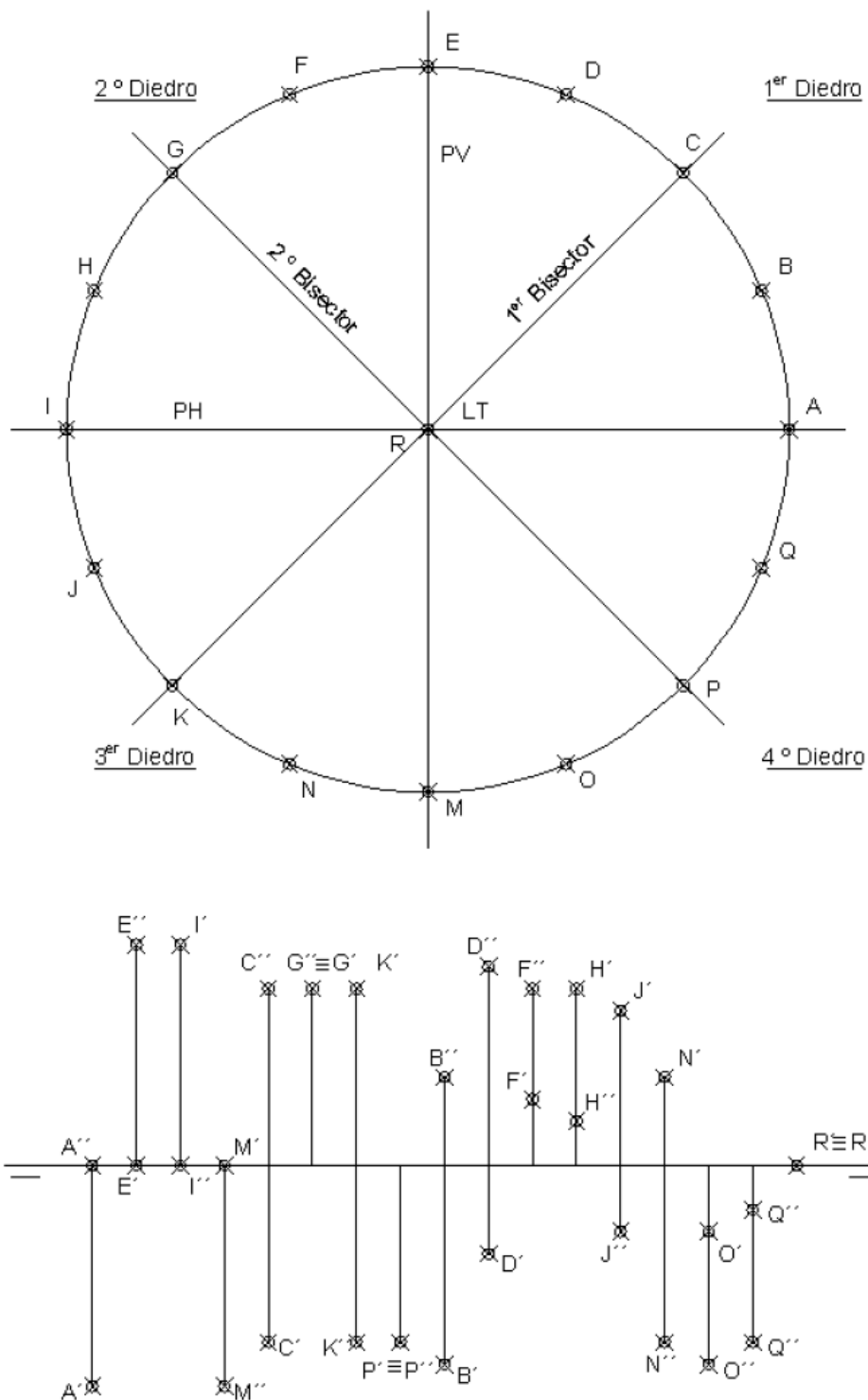


Fig. 4.6 Alfabeto del Punto

En la Fig. 4.6 se ven todas las posibles posiciones del punto y las características de cada uno de ellos:

- Los puntos B, C y D están situados en el **Primer Cuadrante** en su representación siempre tienen la proyección vertical encima de la Línea de Tierra (Z positiva) y la proyección horizontal debajo de la Línea de Tierra (Y positiva).
 - El punto B está situado en el **Primer Octante**, los puntos situados en este Octante también lo estarán en el Primer Cuadrante por lo que tendrá las mismas propiedades que estos (Y y Z positivas), además de que el alejamiento ($|Y|$) será mayor que la cota ($|Z|$).
 - El punto D está situado en el **Segundo Octante**, los puntos situados en este Octante también lo estarán en el Primer Cuadrante por lo que tendrá las mismas propiedades que estos (Y y Z positivas), además de que la cota ($|Z|$) será mayor que el alejamiento ($|Y|$).
- Los puntos F, G y H están situados en el **Segundo Cuadrante** en su representación siempre tienen la proyección vertical y la proyección horizontal encima de la Línea de Tierra (Y negativa y Z positiva).
 - El punto F está situado en el **Tercer Octante**, los puntos situados en este Octante también lo estarán en el Segundo Cuadrante por lo que tendrá las mismas propiedades que estos (Y negativa y Z positiva), además de que de la cota ($|Z|$) será mayor que el alejamiento ($|Y|$).
 - El punto H está situado en el **Cuarto Octante**, los puntos situados en este Octante también lo estarán en el Segundo Cuadrante por lo que tendrá las mismas propiedades que estos (Y negativa y Z positiva), además de que el alejamiento ($|Y|$) será mayor que la cota ($|Z|$).
- Los puntos J, K y N están situados en el **Tercer Cuadrante** en su representación siempre tienen la proyección vertical debajo de la Línea de Tierra (Z negativa) y la proyección horizontal encima de la Línea de Tierra (Z negativa).
 - El punto J está situado en el **Quinto Octante**, los puntos situados en este Octante también lo estarán en el Tercer Cuadrante por lo que tendrá las mismas propiedades que estos (Y y Z negativas), además de que el alejamiento ($|Y|$) será mayor que la cota ($|Z|$).
 - El punto N está situado en el **Sexto Octante**, los puntos situados en este Octante también lo estarán en el Tercer Cuadrante por lo que tendrá las mismas propiedades que estos (Y y Z negativas), además de que la cota ($|Z|$) será mayor que el alejamiento ($|Y|$).
- Los puntos O, P y Q están situados en el **Cuarto Cuadrante** en su representación siempre tienen la proyección vertical y la proyección horizontal debajo de la Línea de Tierra (Y positiva y Z negativa).
 - El punto O está situado en el **Septimo Octante**, los puntos situados en este Octante también lo estarán en el Cuarto Cuadrante por lo que tendrá las mismas propiedades que estos (Y positiva y Z negativa), además de que el alejamiento ($|Y|$) será mayor que la cota ($|Z|$).
 - El punto Q está situado en el **Octavo Octante**, los puntos situados en este Octante también lo estarán en el Cuarto Cuadrante por lo que tendrá las mismas propiedades que estos (Y positiva y Z negativa), además de que la cota ($|Z|$) será mayor que el alejamiento ($|Y|$).

- Los puntos A y I están situados en el **Plano Horizontal** de proyección siempre tienen cota (Z) cero.
- Los puntos E y M están situados en el **Plano Vertical** de proyección siempre tienen alejamiento (Y) cero.
- Los puntos C, G, K, P están situados en los planos bisectores, siempre tienen iguales la cota y el alejamiento ($|Y| = |Z|$).
- El punto R está situado en la Línea de Tierra, estos puntos siempre tienen cota y alejamiento cero.

4.3. Recta

4.3.1. Representación de la recta

La recta es uno de los elementos geométricos básicos. En el sistema diédrico las rectas del espacio se representan mediante sus proyecciones ortogonales sobre los planos coordenados. Por ejemplo en la Fig. 4.7 la recta r con una proyección vertical (r''), en el Plano de Proyección Vertical, y otra proyección horizontal (r'), en el Plano de Proyección Horizontal. En general, la proyección de una recta es otra recta.

Cualquier recta situada en el espacio sin condiciones especiales como la de la Fig. 4.7, al prolongarse atraviesa el Plano de proyección Vertical (PV) y el Plano de proyección Horizontal (PH) y pasa por tres diedros o cuadrantes. Los puntos por los que la recta atraviesa los planos de proyección se denominan trazas de la recta. Por tanto las trazas de una recta son los puntos de intersección de la recta con los planos coordenados, el punto de intersección de la recta con el Plano de proyección Horizontal, será la traza horizontal (H). El punto de intersección de la recta con el Plano de proyección Vertical, es la traza vertical (V). La traza vertical de una recta siempre es un punto del Plano de proyección Vertical y la traza horizontal de una recta siempre es un punto del Plano de proyección Horizontal.

En la figura Fig. 4.7 se muestra la representación de una recta r en el Sistema Diédrico, a partir de sus proyecciones, así como sus trazas.

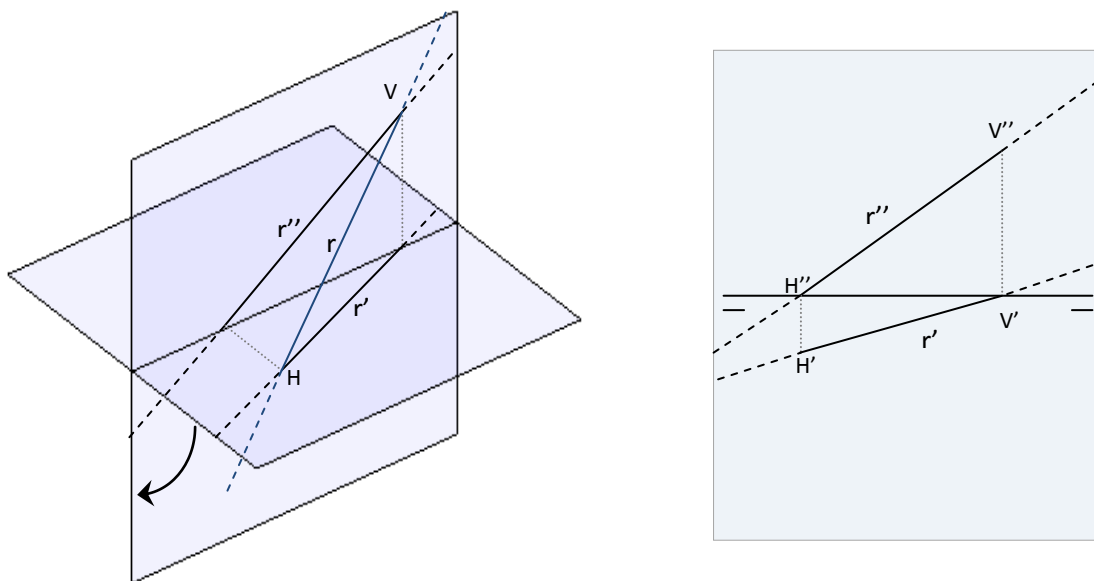


Fig. 4.7 Representación de la Recta

Para dibujar las rectas, se podría considerar que los planos de proyección son semiopacos y que la recta se ve desde el primer diedro, por lo tanto, todo lo que se encuentre en el primer diedro es visible y todo lo que se encuentre en los otros tres cuadrantes está oculto. Las líneas visibles se dibujan con línea continua de espesor grueso. Las líneas ocultas se dibujan con línea a trazos de espesor medio.

Para saber qué zonas de una recta están en distintos diedros, es preciso localizar primero las trazas de la recta, ya que estos puntos marcan los puntos donde la recta cambia de diedro. La zona de la recta situada entre las trazas pertenece a un determinado diedro. La zona de la recta que va desde la traza de la izquierda hacia la izquierda pertenece a otro diedro. La zona de la recta que va desde la traza de la derecha hacia la derecha pertenece a otro diedro.

Para saber a qué diedro o cuadrante pertenece una determinada zona, es suficiente con elegir un punto de cada zona que pertenezca a la recta y ver a qué diedro pertenece dicho punto.

Otra forma de saber a qué cuadrante pertenece cada zona es fijándose en las trazas. En la Tabla 4.1 se resume la visibilidad de la recta según sus trazas.

Tabla 4.1 Visibilidad de las Rectas según sus Trazas

1.	-----	$H^{(+)}$	-----	$V^{(+)}$	-----
2.	-----	$H^{(+)}$	-----	$V^{(-)}$	-----
3.	-----	$H^{(-)}$	-----	$V^{(+)}$	-----
4.	-----	$H^{(-)}$	-----	$V^{(-)}$	-----

En la Tabla 4.1 se considera proyección vertical positiva y se representa como $V^{(+)}$ cuando su tercera coordenada (z) sea positiva, si es negativa será $V^{(-)}$. En el caso de la proyección

horizontal se considera positiva cuando su segunda coordenada (y) sea positiva y se representa $H^{(+)}$, si es negativa será $H^{(-)}$.

Según la Tabla 4.1 el segmento correspondiente al primer diedro teniendo en cuenta las proyecciones será:

1. Las dos proyecciones positivas: Segmento que las une.
2. Proyección horizontal positiva y la vertical negativa: semirrecta que va de la proyección horizontal en el sentido contrario al de la proyección vertical
3. Proyección horizontal negativa y la vertical positiva: semirrecta que va de la proyección vertical en el sentido contrario al de la proyección horizontal.
4. Las dos proyecciones negativas: Ningún segmento en el primer diedro.

Para el caso de que las dos proyecciones coincidan, no contemplado en la Tabla 4.1, será necesario un punto distinto a este para saber que diedros ocupa.

Una recta está definida por dos puntos. Con las proyecciones de dos puntos cualesquiera, se pueden conseguir las proyecciones de la recta definida por dichos puntos como se muestra en la Fig. 4.8, uniendo sus proyecciones homónimas, esto es, por un lado las proyecciones verticales de los dos puntos dando lugar a la proyección vertical de la recta y por otro lado las proyecciones horizontales de los dos puntos dando lugar a la proyección horizontal de la recta, teniendo la visibilidad.

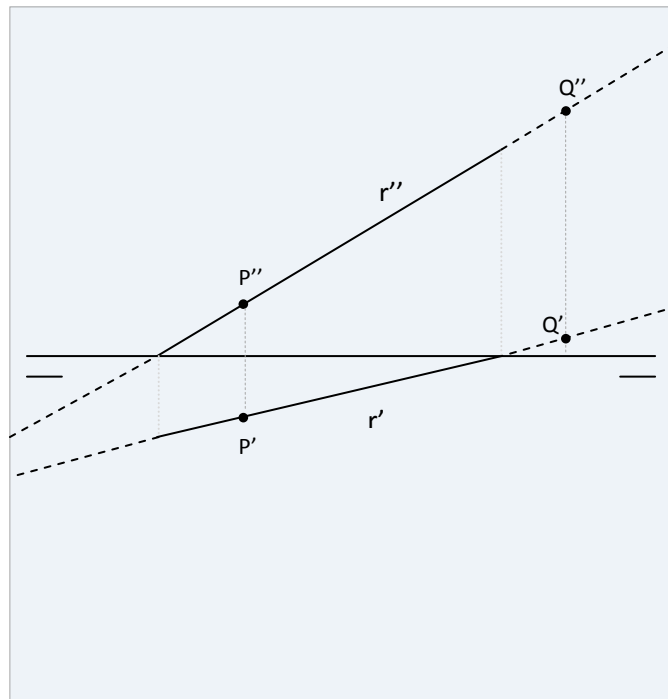


Fig. 4.8 Representación de la Recta a partir de dos Puntos

Como ocurre en la Fig. 4.8 con los puntos P y Q y la recta r, un punto pertenece a una recta si su proyección vertical, en este caso P'' y Q'' , está en la proyección vertical de la recta, r'' , y si su proyección horizontal, P' y Q' , está en la proyección horizontal de la recta, r' . No es suficiente con que solo coincida una de las proyecciones.

Se dice que una recta es una sucesión de puntos en la misma dirección. Técnicamente es el lugar geométrico de los puntos que tienen una misma dirección.

Como en la Fig. 4.7 para definir una recta totalmente es preciso indicar las siguientes características:

- Las proyecciones
- Las trazas
- Los cuadrantes por los que pasa
- Las partes vistas y ocultas

4.3.2. Ecuaciones de la recta

Una recta r en el espacio queda determinada conociendo un punto P y un vector \vec{v} no nulo que se llama vector director o direccional de la recta.

$$P_0 = (x_0, y_0, z_0) \quad \vec{v} = (v_x, v_y, v_z)$$

Existen diferentes formas de expresar la ecuación de una recta:

Ecuación en forma vectorial

La recta r que pasa por el punto P_0 y tiene como vector director \vec{v} es el conjunto de puntos del espacio que verifican la siguiente relación vectorial:

$$\overrightarrow{P_0P} = \lambda \vec{v} \quad \text{con} \quad \lambda \in \mathbb{R} \quad \text{teniendo en cuenta que} \quad P = P_0 + \overrightarrow{P_0P}, \text{ resulta:}$$

$$P = P_0 + \lambda \vec{v}$$

Ecuación en forma paramétrica

Desarrollando la ecuación vectorial anterior expresada en coordenadas, se tiene lo siguiente:

$$(x, y, z) = (x_0, y_0, z_0) + (\lambda v_x, \lambda v_y, \lambda v_z) = (x_0 + \lambda v_x, y_0 + \lambda v_y, z_0 + \lambda v_z)$$

Igualando componentes resulta:

$$r: \begin{cases} x = x_0 + \lambda v_x \\ y = y_0 + \lambda v_y \\ z = z_0 + \lambda v_z \end{cases}$$

Ecuación de forma continua

Si, en las ecuaciones paramétricas, v_x , v_y y v_z son distintos de cero, se puede despejar en cada una de ellas el parámetro λ :

$$\lambda = \frac{x - x_0}{v_x}; \quad \lambda = \frac{y - y_0}{v_y}; \quad \lambda = \frac{z - z_0}{v_z}$$

Igualando las expresiones obtenidas resulta:

$$r : \frac{x - x_0}{v_x} = \frac{y - y_0}{v_y} = \frac{z - z_0}{v_z}$$

Ecuación en forma general, cartesiana o implícita

A partir de la ecuación forma continua de la recta se pueden obtener las dos ecuaciones siguientes:

$$\frac{x - x_0}{v_x} = \frac{y - y_0}{v_y}$$

$$\frac{y - y_0}{v_y} = \frac{z - z_0}{v_z}$$

que se pueden reescribir de la forma:

$$r: \begin{cases} ax + by + cz + d = 0 \\ a'x + b'y + c'z + d' = 0 \end{cases}$$

4.3.3. Posiciones particulares de la recta

1. Recta Inclinada u Oblicua

Como se muestra en Fig. 4.9 las rectas oblicuas no tienen ninguna condición especial, pasan por tres diedros, cortando por los dos planos de proyección por lo que tiene dos trazas.

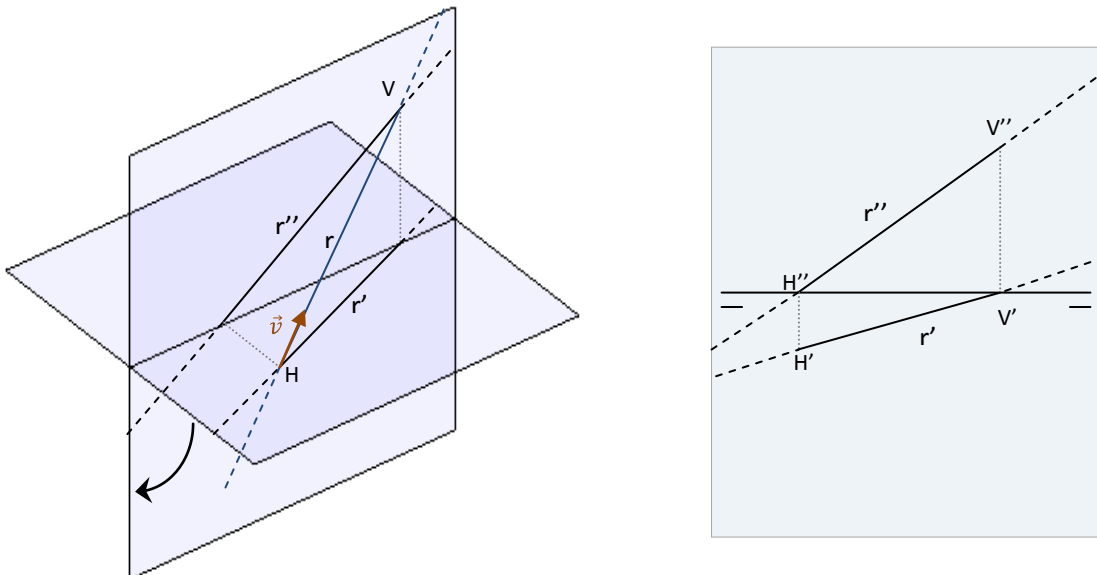


Fig. 4.9 Recta Inclinada u Oblicua

La recta oblicua no tiene ninguna característica especial, su vector director no tendrá ningún elemento nulo:

$$\vec{v} = (v_x, v_y, v_z)$$

2. Recta Horizontal

Como aparece en la Fig. 4.10 una recta horizontal es paralela al Plano de proyección Horizontal, por lo que no corta en él y no tendrá traza horizontal, es decir estas rectas sólo tienen una traza. Además su proyección vertical es paralela a la Línea de Tierra y su proyección horizontal se muestra en verdadera magnitud. Pasa por dos cuadrantes.

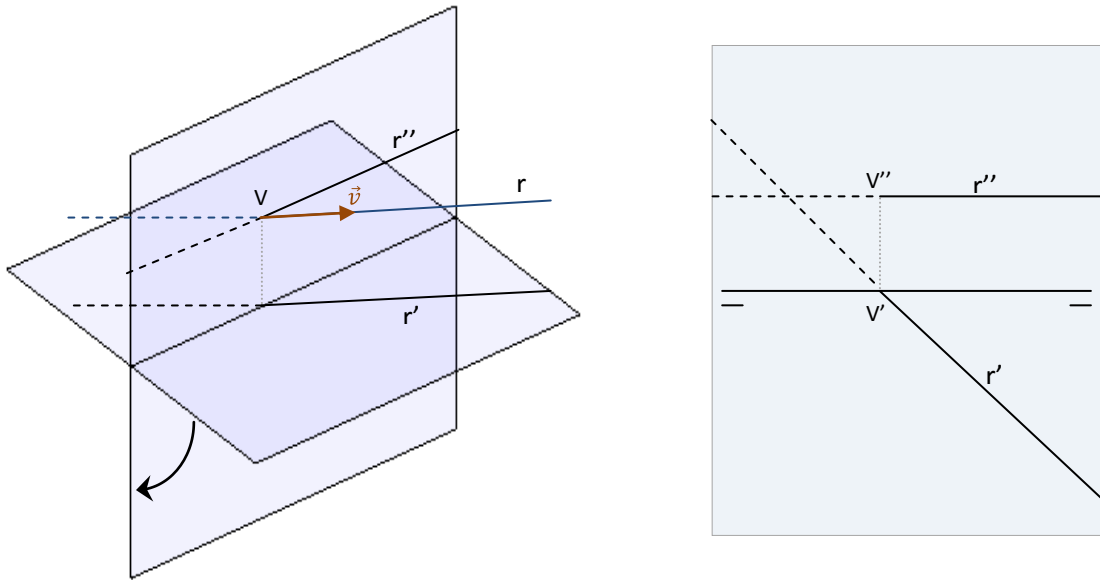


Fig. 4.10 Recta Horizontal

En las rectas horizontales la característica principal es que la Z es constante y al ser el vector director la diferencia entre dos de los puntos de la recta, la coordenada Z de este será 0:

$$\vec{v} = (v_x, v_y, 0)$$

3. Recta Frontal

Como aparece en la Fig. 4.11 una Recta frontal es paralela al Plano de proyección Vertical, por lo que no corta en él y no tendrá traza vertical, es decir estas rectas sólo tienen una traza. Además su proyección horizontal es paralela a la Línea de Tierra y su proyección vertical se muestra en verdadera magnitud. Pasa por dos cuadrantes.

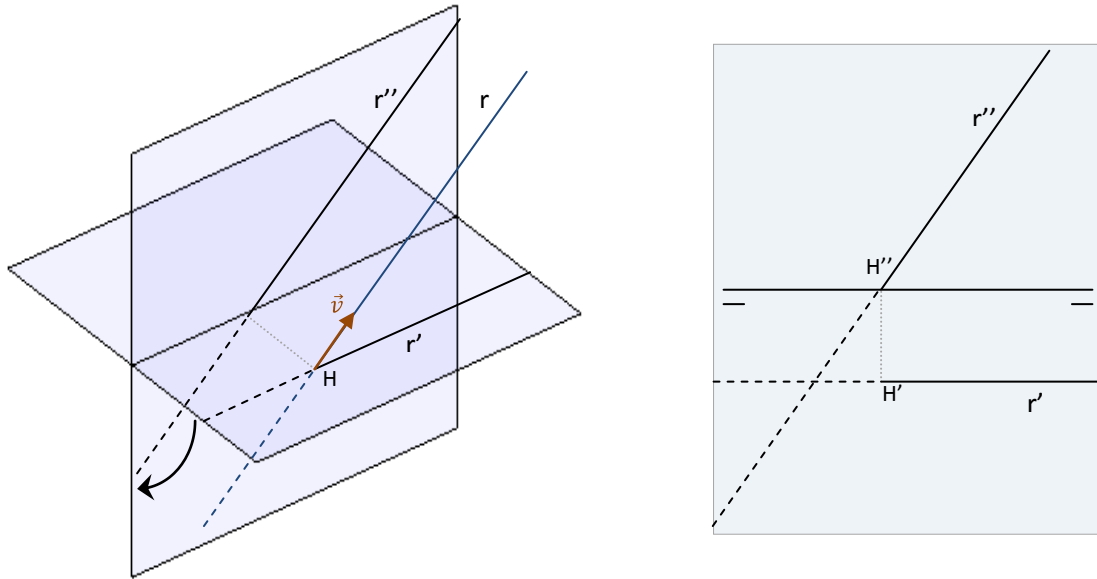


Fig. 4.11 Recta Frontal

En las rectas frontales la característica principal es que la Y es constante y al ser el vector director la diferencia entre dos de los puntos de la recta, la coordenada Y de este será 0:

$$\vec{v} = (v_x, 0, v_z)$$

4. Recta de Punta

En la Fig. 4.12 se muestra la recta de punta que es perpendicular al Plano de proyección Vertical. Su proyección horizontal es perpendicular a la Línea de Tierra y su proyección vertical es un punto que coincide con su traza vertical, que es la única que tiene ya que es una recta paralela al Plano de proyección Horizontal, por lo que además su proyección horizontal se muestra en verdadera magnitud. Tiene una traza. Pasa por dos cuadrantes.

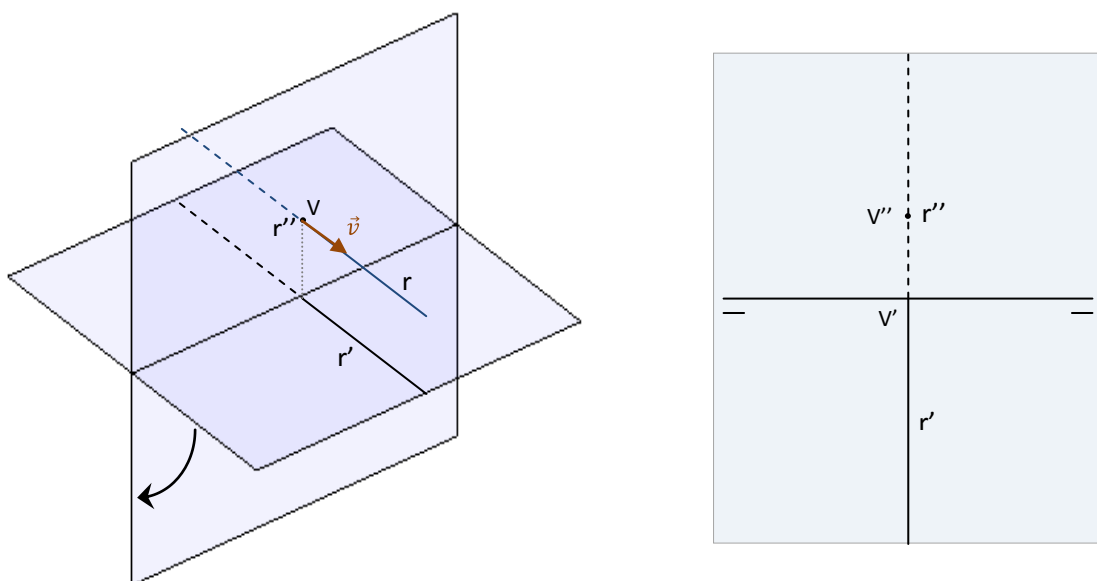


Fig. 4.12 Recta de Punta

En las rectas de punta la característica principal es que la X y la Z son constantes y al ser el vector director la diferencia entre dos de los puntos de la recta, las coordenadas X y Z de este serán 0:

$$\vec{v} = (0, v_y, 0)$$

5. Recta Vertical

En la Fig. 4.13 se muestra la Recta Vertical que es perpendicular al Plano de proyección Horizontal. Su proyección vertical es perpendicular a la Línea de Tierra y su proyección horizontal es un punto que coincide con su traza horizontal, que es la única ya que es una recta paralela al Plano de proyección Vertical, por lo que además su proyección vertical se muestra en verdadera magnitud. Pasa por dos cuadrantes.

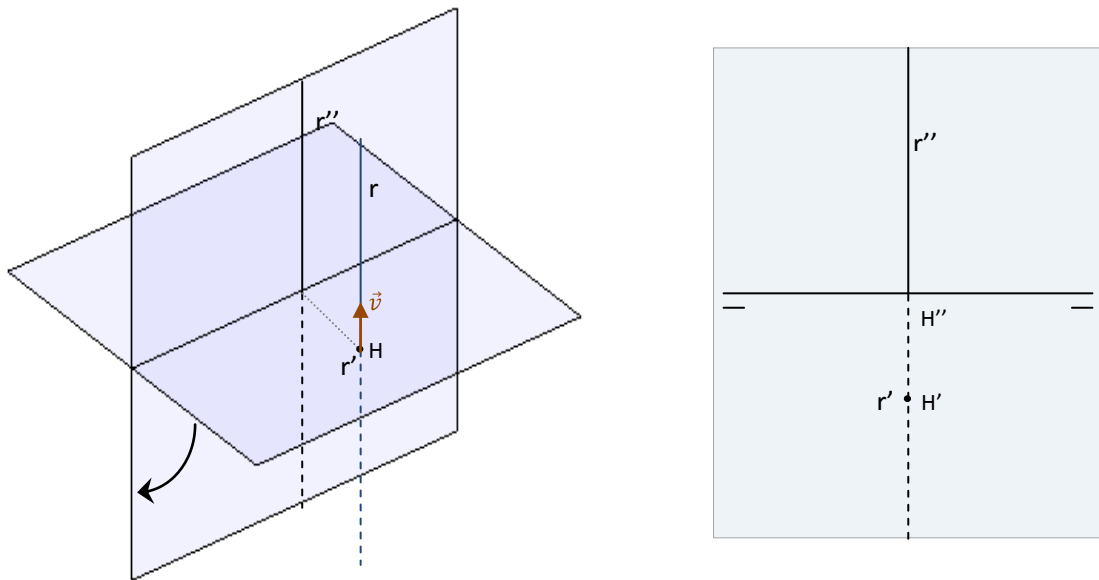


Fig. 4.13 Recta Vertical

En las rectas verticales la característica principal es que la X y la Y son constantes y al ser el vector director la diferencia entre dos de los puntos de la recta, las coordenadas X y Y de este serán 0:

$$\vec{v} = (0, 0, v_z)$$

6. Recta Paralela a la LT

En la Fig. 4.14 se muestra una recta paralela a la Línea de Tierra. Sus dos proyecciones son paralelas a Línea de Tierra por lo que sus dos proyecciones se muestran en verdadera magnitud y no tiene trazas. Pasa por un cuadrante.

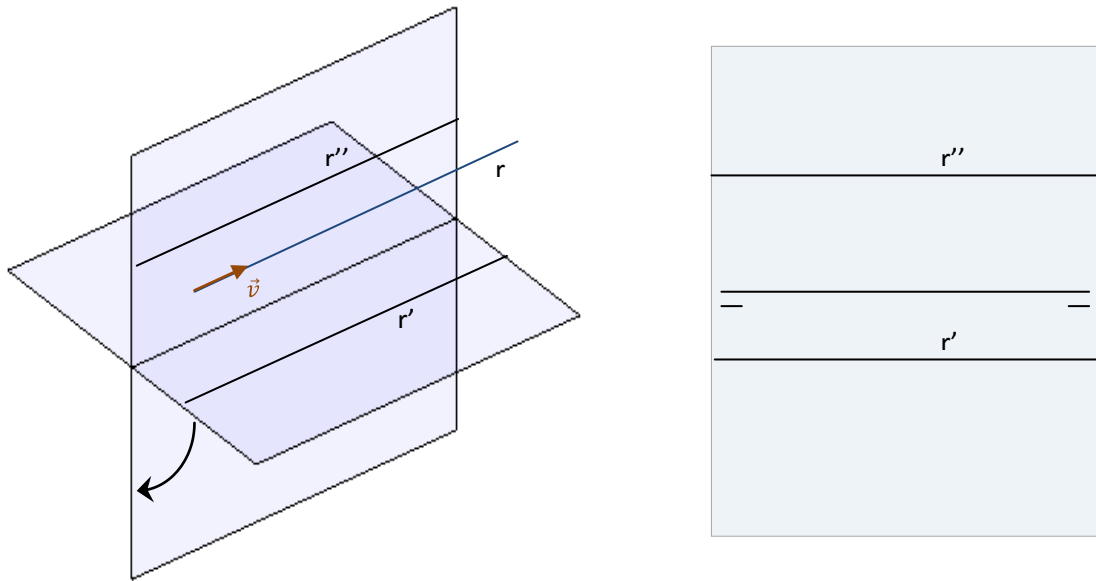


Fig. 4.14 Recta Paralela a LT

En las rectas paralelas a la Línea de Tierra la característica principal es que la Y y la Z son constantes y al ser el vector director la diferencia entre dos de los puntos de la recta, las coordenadas Y y Z de éste serán 0:

$$\vec{v} = (v_x, 0, 0)$$

7. Recta que Corta a la LT

En la Fig. 4.15 se muestra una recta que corta a la Línea de Tierra en un punto. Sus dos proyecciones son inclinadas con la Línea de Tierra y se cortan en un punto de ella. Tiene dos trazas coincidentes en un punto de Línea de Tierra. Pasa por dos cuadrantes.

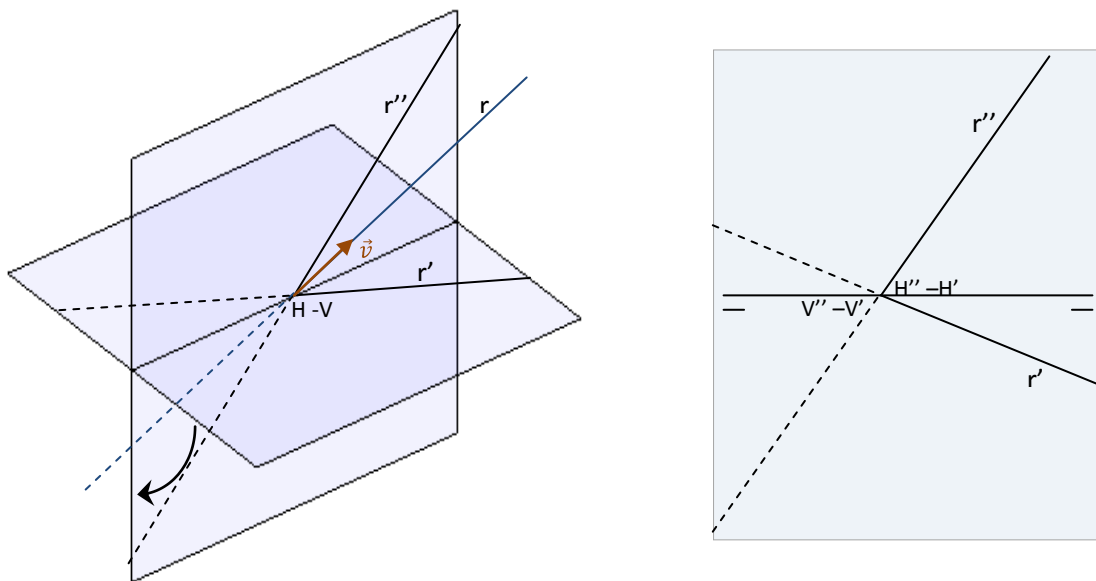


Fig. 4.15 Recta que Corta a LT

La característica principal de las rectas que cortan a Línea de Tierra es que sus trazas vertical y horizontal coinciden:

$$H \equiv V$$

8. Recta de Perfil

En la Fig. 4.16 se muestra una recta de perfil que está contenida en un plano de perfil. Sus dos proyecciones son perpendiculares a Línea de Tierra. Las proyecciones vertical y horizontal se complementan con una proyección de perfil. Existen varios subtipos de rectas de perfil con características propias.

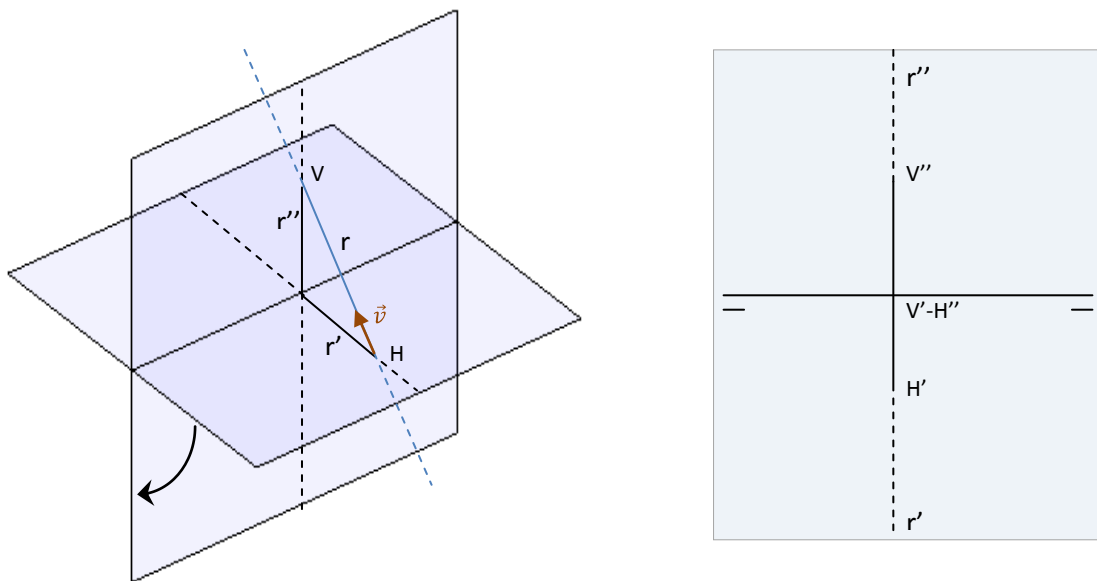


Fig. 4.16 Recta de Perfil

En las rectas de perfil la característica principal es que la X es constante y al ser el vector director la diferencia entre dos de los puntos de la recta, la coordenadas X de este será 0:

$$\vec{v} = (0, v_y, v_z)$$

En la Tabla 4.2 se observa las relaciones que hay entre los distintos tipos de rectas y sus vectores directores, marcándose con un punto gordo las trazas en caso de tenerlas, y mostrándose solo la parte visible, es decir la del primer diedro.

Tabla 4.2 Rectas vs Vectores Directores

TIPO RECTA	DIBUJO	RELACIONES
Oblicua		Sin relación
Horizontal		$v_z = 0$
Frontal		$v_y = 0$
De Punta		$v_x = 0$ $v_z = 0$
Vertical		$v_x = 0$ $v_y = 0$
Paralela a LT		$v_y = 0$ $v_z = 0$
Corta a LT		$H \equiv V$
Perfil		$v_x = 0$

4.3.4. Posiciones relativas de dos rectas

Dos rectas en el espacio pueden estar contenidas en un mismo plano, o no, según sea su posición relativa. Dos rectas solo pueden tener tres clases de posiciones relativas entre sí:

Rectas que se cruzan

Se denominan rectas que se cruzan a las rectas que no están contenidas en el mismo plano, y por lo tanto, no tienen ningún punto en común ni son paralelas. Son rectas sin ninguna condición especial.

Rectas que se cortan

Se dice que dos rectas se cortan en el espacio si tienen un punto común. Estas rectas siempre definen un plano.

Rectas paralelas

Se dice que dos rectas son paralelas en el espacio si tienen la misma dirección. Estas rectas siempre definen un plano.

Dos rectas son paralelas si sus vectores directores son paralelos, es decir, si éstos son linealmente dependientes.

$$\vec{u} = k\vec{v}$$

4.4. Plano

4.4.1. Representación del plano

Todo plano en el espacio del Sistema Diédrico se representa con las dos trazas que se generan automáticamente al cortar con los planos de proyección, dando lugar a dos rectas: una traza vertical, en el Plano de proyección Vertical (α'), y otra traza horizontal (α''), en el Plano de proyección Horizontal, como el plano α de ejemplo de la Fig. 4.17.

Un plano siempre es ilimitado, y por lo tanto, sus trazas también son ilimitadas.

Las trazas de un plano dispuesto sin ninguna condición especial se cortan en un mismo punto de la Línea de Tierra.

En la Fig. 4.17 se muestra la representación de un plano α en el Sistema Diédrico, a partir de sus trazas, que cortan en un punto de la línea de tierra.

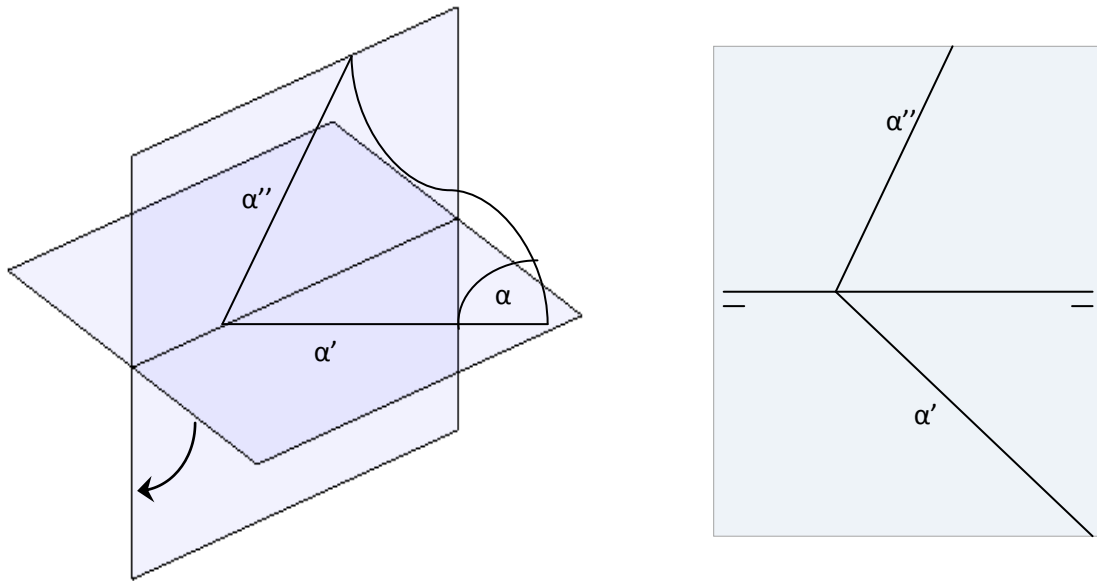


Fig. 4.17 Representación del Plano

Un plano no tiene proyección (la proyección de los infinitos puntos que lo componen daría como resultado una mancha de puntos), por este motivo los planos se representan en el Sistema Diédrico por sus trazas. Una recta pertenece a un plano, o está contenida en un plano, si las trazas de la recta coinciden con las trazas del plano. Además un punto pertenece a un plano, o está contenido en un plano, si pertenece a cualquier recta del plano.

Un plano puede estar definido por:

- Tres puntos no alineados.
Los puntos estarán alineados si forman parte de la misma recta.
- Una recta y un punto exterior a ella.
- Dos rectas paralelas.
Dos rectas serán paralelas si sus vectores directores son proporcionales (y no tiene puntos coincidentes, en tal caso sería la misma recta).
- Dos rectas que se cortan.
Dos rectas que se cortan tendrán un punto en común (el punto de corte).

Para representar un plano en el Sistema Diédrico con los elementos que contiene es necesario dos rectas, de tal forma que si tenemos 3 puntos se sacarán dos rectas a partir de éstos, y con un punto y una recta se puede dibujar una recta que pase por el punto y corte a la recta ya representada, caso de la Fig. 4.18. Una vez que tenemos dos rectas, en la Fig. 4.18 serían las rectas r y s , uniremos las proyecciones horizontales de las dos trazas horizontales de las rectas, Hr' y Hs' , formando la traza horizontal del plano, α' , y las proyecciones verticales de las dos trazas verticales de las rectas, Vr'' y Vs'' , formando la traza vertical del plano, α'' .

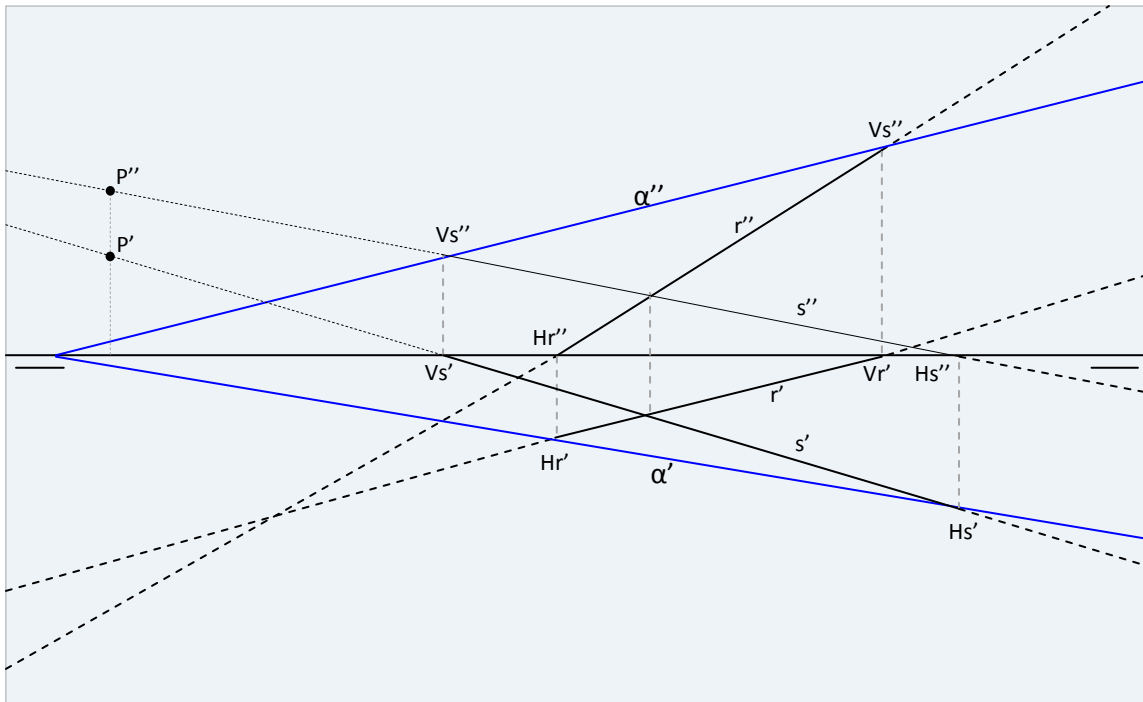


Fig. 4.18 Representación del Plano a partir de un Punto y una Recta

4.4.2. Ecuaciones del plano

Un plano π queda determinado cuando se conoce un punto P_0 del mismo y dos vectores \vec{u} y \vec{v} no nulos y linealmente independientes que están contenidos en el plano, llamados vectores directores del plano.

$$P_0 = (x_0, y_0, z_0) \quad \vec{u} = (u_x, u_y, u_z) \quad \vec{v} = (v_x, v_y, v_z)$$

Existen diferentes formas de expresar la ecuación de un plano. Se describen a continuación.

Ecuación en forma vectorial

El plano π que contiene al punto P_0 y tiene como vectores directores los vectores \vec{u} y \vec{v} es el conjunto de puntos del espacio que verifican la siguiente relación vectorial:

$$\overrightarrow{P_0P} = \lambda \vec{u} + \mu \vec{v} \quad \text{con} \quad \lambda, \mu \in \mathbb{R} \quad \text{teniendo en cuenta que} \quad P = P_0 + \overrightarrow{P_0P}, \text{ resulta:}$$

$$P = P_0 + \lambda \vec{u} + \mu \vec{v}$$

Ecuación en forma paramétrica

Desarrollando la ecuación vectorial expresada en componentes, resulta:

$$\pi: \begin{cases} x = x_0 + \lambda u_x + \mu v_x \\ y = y_0 + \lambda u_y + \mu v_y \\ z = z_0 + \lambda u_z + \mu v_z \end{cases}$$

Ecuación en forma general, cartesiana o implícita

Como $P = P_0 + \lambda \vec{u} + \mu \vec{v}$ en el determinante

$$\begin{vmatrix} x - x_0 & u_x & v_x \\ y - y_0 & u_y & v_y \\ z - z_0 & u_z & v_z \end{vmatrix}$$

La primera columna es combinación lineal de la segunda y de la tercera. Por tanto dicho determinante es cero. Desarrollando el determinante, agrupando términos e igualando a 0, nos queda una ecuación de la forma:

$$\pi: ax + by + cz + d$$

La relación con la ecuación vectorial se obtiene desarrollando el determinante:

$$(x - x_0)(u_y v_z - u_z v_y) + (y - y_0)(u_z v_x - u_x v_z) + (z - z_0)(u_x v_y - u_y v_x) = a(x - x_0) + b(y - y_0) + c(z - z_0) = ax + by + cz + d$$

Quedaría:

$$a = (u_y v_z - u_z v_y)$$

$$b = (u_z v_x - u_x v_z)$$

$$c = (u_x v_y - u_y v_x)$$

$$d = -ax_0 - by_0 - cz_0$$

Ecuación normal

Otra forma de determinar la ecuación de un plano es conociendo un punto del mismo y un vector normal al plano.

Sea $P_0 = (x_0, y_0, z_0)$ un punto dado del plano π y sea $\vec{n} = (a, b, c)$ un vector normal a π . Entonces, para cualquier punto $P = (x, y, z)$ del plano π el vector $\overrightarrow{P_0 P}$ es perpendicular a \vec{n} , de manera que:

$$\vec{n} \cdot \overrightarrow{P_0 P} = 0$$

A partir de la ecuación normal del plano se puede obtener muy fácilmente su ecuación general:

$$\vec{n} \cdot \overrightarrow{P_0 P} = a(x - x_0) + b(y - y_0) + c(z - z_0) = 0 \rightarrow ax + by + cz + d = 0$$

4.4.3. Posiciones particulares del plano

Los planos tienen posiciones relativas con respecto a los planos proyectantes y a la Línea de Tierra.

9. Plano Oblicuo o Inclinado

En la Fig. 4.19 se representa plano oblicuo, estos planos no tienen ninguna condición especial, cortan a los dos planos de proyección con ángulos desiguales y sus trazas son oblicuas a la Línea de Tierra cortándola siempre en un punto.

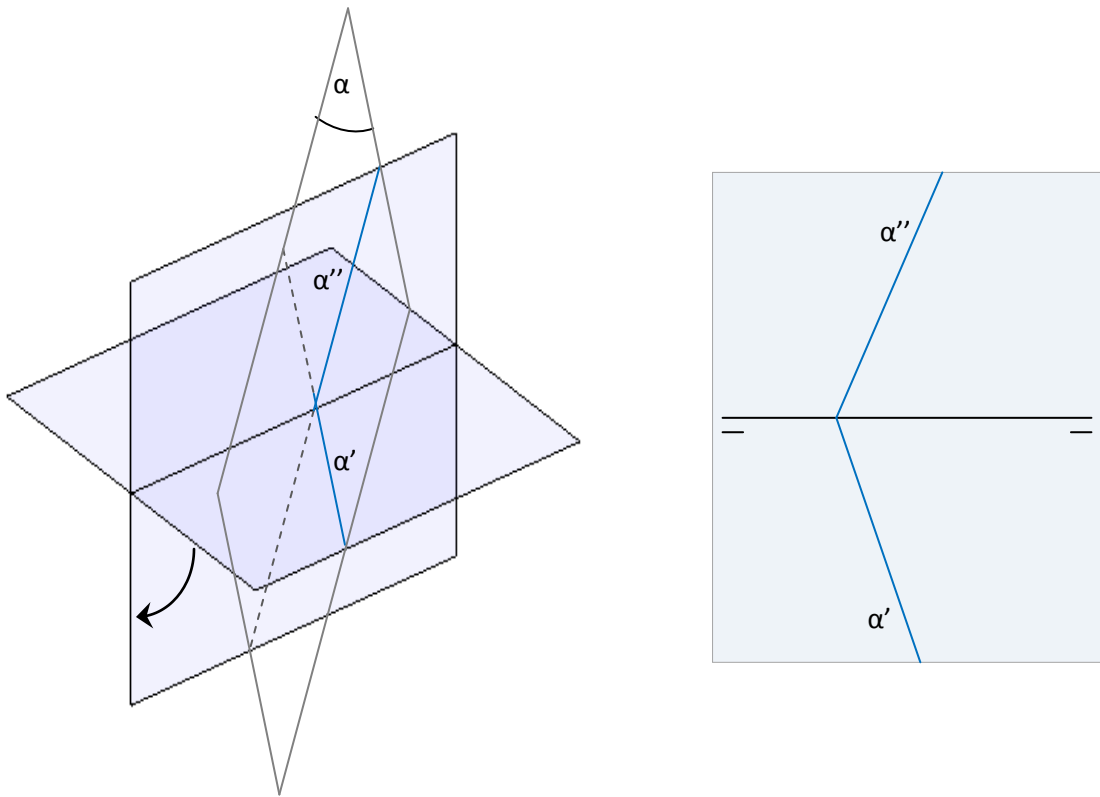


Fig. 4.19 Plano Oblicuo

El plano oblicuo no tiene ninguna característica especial, en la ecuación general no aparecerá ningún elemento necesariamente igual a 0, y no habrá ninguna relación en sus vectores directores.

$$\vec{u} = (u_x, u_y, u_z) \quad ax + by + cz + d = 0$$

$$\vec{v} = (v_x, v_y, v_z)$$

10. Plano Horizontal

Como se muestra en la Fig. 4.20 un plano horizontal es paralelo al Plano de proyección Horizontal por lo que sólo tiene traza vertical, que es paralela a Línea de Tierra. Pasa por dos cuadrantes.

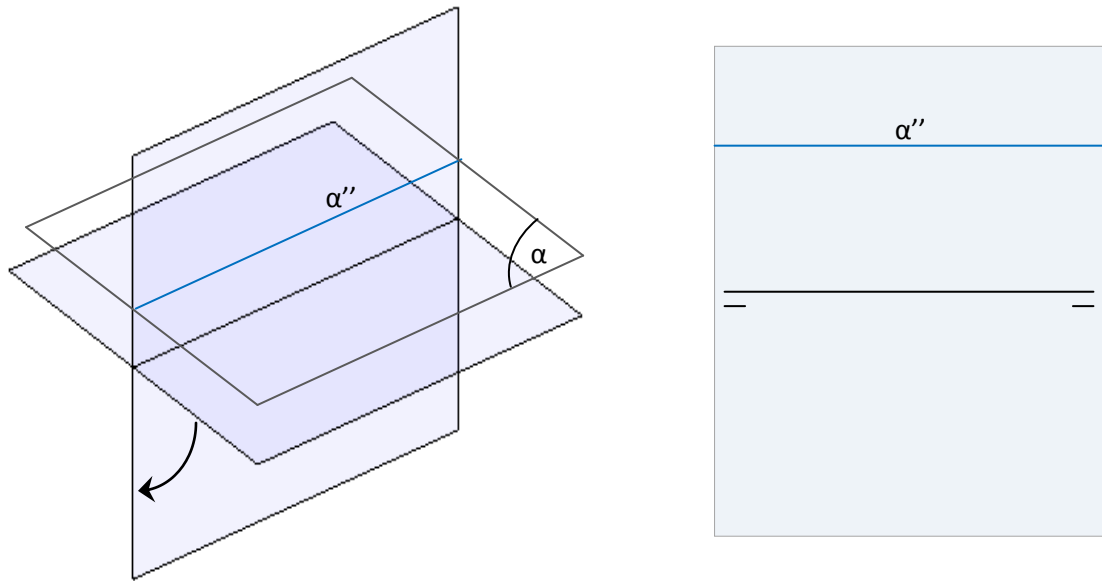


Fig. 4.20 Plano Horizontal

En los planos horizontales la característica principal es que la Z es constante con lo que tenemos que la ecuación general será:

$$cz + d = 0$$

Y al ser los vectores directores la diferencia entre dos de los puntos del plano, la coordenada Z de estos será 0:

$$\vec{u} = (u_x, u_y, 0)$$

$$\vec{v} = (v_x, v_y, 0)$$

11. Plano Frontal

Como se muestra en la Fig. 4.21 el plano frontal es paralelo al Plano de proyección Vertical por lo que sólo tiene traza horizontal, que es paralela a Línea de Tierra. Pasa por dos cuadrantes.

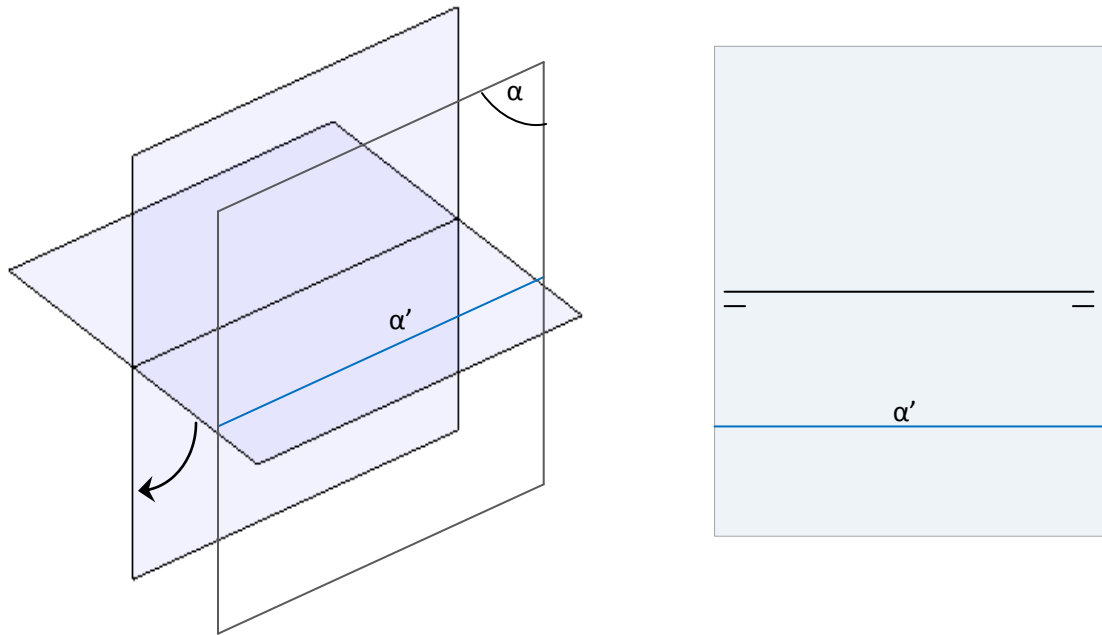


Fig. 4.21 Plano Frontal

En los planos frontales la característica principal es que la Y es constante con lo que se tiene que la ecuación general será:

$$by + d = 0$$

Y al ser los vectores directores la diferencia entre dos de los puntos del plano, la coordenada Y de éstos será 0:

$$\vec{u} = (u_x, 0, u_z)$$

$$\vec{v} = (v_x, 0, v_z)$$

12. Plano de Perfil

Como se muestra en la Fig. 4.22 un plano de perfil es perpendicular a la Línea de Tierra y a los planos proyectantes. Sus dos trazas son perpendiculares a Línea de Tierra. Pasa por cuatro cuadrantes.

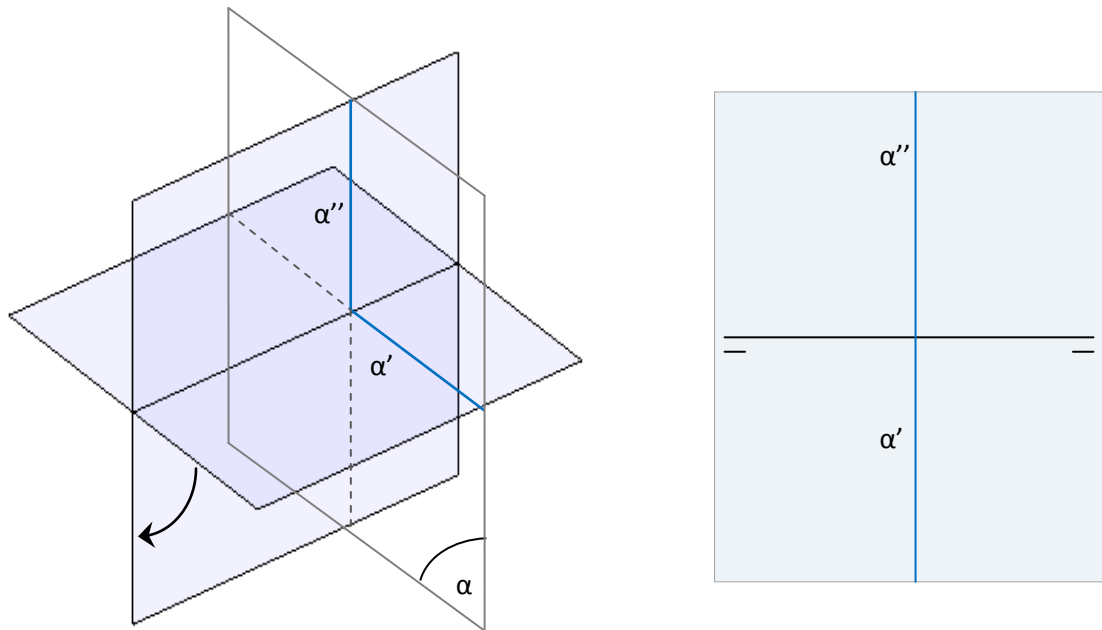


Fig. 4.22 Plano de Perfil

En los planos de perfil la característica principal es que la X es constante con lo que se tiene que la ecuación general será:

$$ax + d = 0$$

Y al ser los vectores directores la diferencia entre dos de los puntos del plano, la coordenada y de estos será 0:

$$\vec{u} = (0, u_y, u_z)$$

$$\vec{v} = (0, v_y, v_z)$$

13. Plano Vertical

Como muestra en la Fig. 4.23 un plano vertical es perpendicular al Plano de proyección Horizontal, por lo que su traza vertical es perpendicular a Línea de Tierra. Pasa por cuatro cuadrantes.

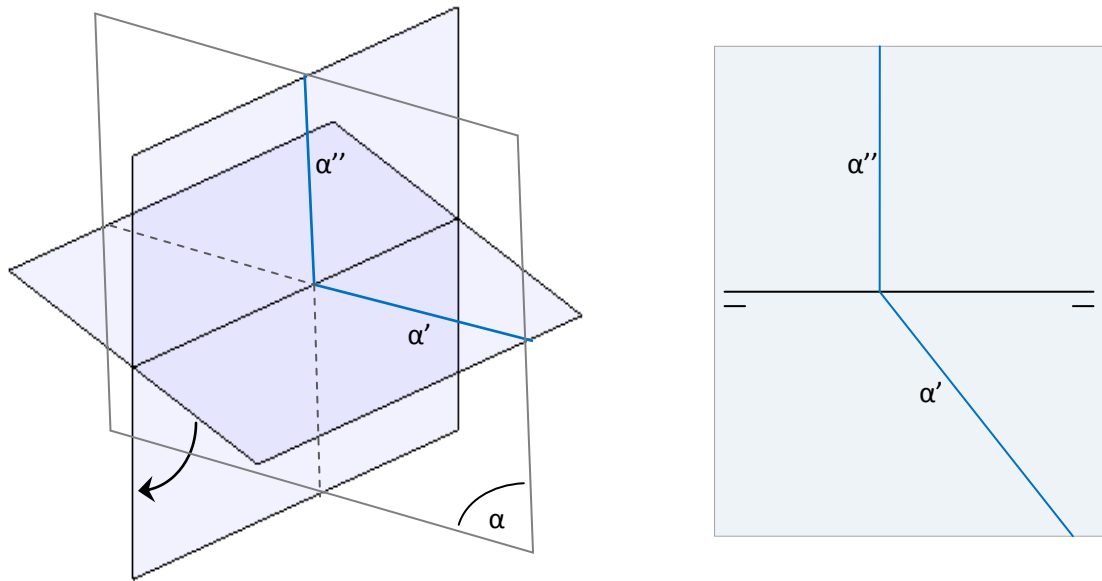


Fig. 4.23 Plano Vertical

En la representación en 3D de un plano vertical de la Fig. 4.23 se observa una X y una Y constante en la traza vertical, dando lugar a una recta perpendicular a la línea de tierra, con lo que uno de los vectores directores podría ser $\vec{u} = (0,0,u_z)$, y teniendo otro vector director $\vec{v} = (v_x, v_y, v_z)$, y el punto del plano $P_0 = (x_0, y_0, z_0)$ la ecuación general será:

$$\begin{vmatrix} x - x_0 & 0 & v_x \\ y - y_0 & 0 & v_y \\ z - z_0 & u_z & v_z \end{vmatrix} = (x - x_0)(-u_z v_y) + (y - y_0)(u_z v_x) = a(x - x_0) + b(y - y_0)$$

$$\text{Siendo } c = 0 \rightarrow c = (u_x v_y - u_y v_x) = 0 \rightarrow u_x v_y = u_y v_x$$

$$ax + by + d = 0$$

$$\vec{u} = (u_x, \lambda u_x, u_z)$$

$$\vec{v} = (v_x, \lambda v_x, v_z)$$

14. Plano de Canto

Como se muestra en la Fig. 4.24 un plano de canto es perpendicular al Plano de proyección Vertical por lo que su traza horizontal es perpendicular a Línea de Tierra. Pasa por cuatro cuadrantes.

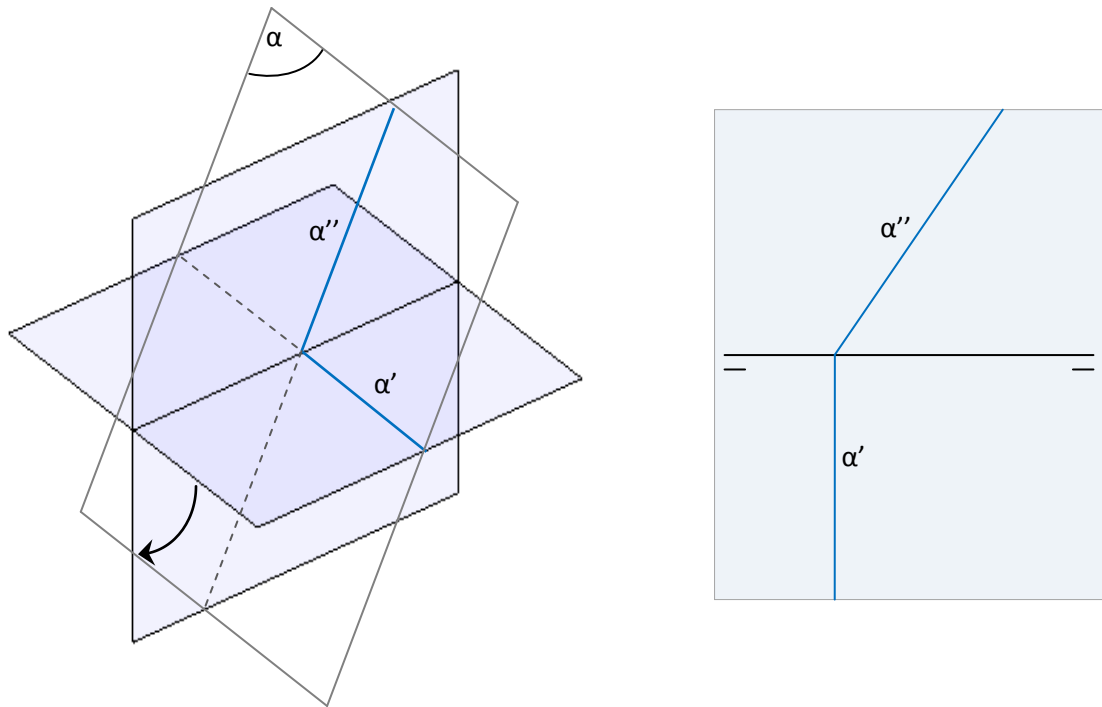


Fig. 4.24 Plano de Canto

El dibujo en 3D de un plano de canto de la Fig. 4.24 se observa una X y una Z constante en la traza horizontal, dando lugar a una recta perpendicular línea de tierra, con lo que uno de los vectores directores podría ser $\vec{u} = (0, u_y, 0)$, y teniendo otro vector director $\vec{v} = (v_x, v_y, v_z)$, y el punto del plano $P_0 = (x_0, y_0, z_0)$ la ecuación general será:

$$\begin{vmatrix} x - x_0 & 0 & v_x \\ y - y_0 & u_y & v_y \\ z - z_0 & 0 & v_z \end{vmatrix} = (x - x_0)(u_y v_z) + (z - z_0)(-u_y v_x) = a(x - x_0) + c(z - z_0)$$

$$\text{Siendo } b = 0 \rightarrow b = (u_z v_x - u_x v_z) \rightarrow u_z v_x = u_x v_z$$

$$ax + cz + d = 0$$

$$\vec{u} = (u_x, u_y, \lambda u_x)$$

$$\vec{v} = (v_x, v_y, \lambda v_x)$$

15. Plano Paralelo a LT

Como se muestra en la Fig. 4.25 un plano paralelo a la Línea de Tierra es oblicuo a los planos proyectantes. Sus dos trazas son paralelas a Línea de Tierra. Pasa por tres cuadrantes.

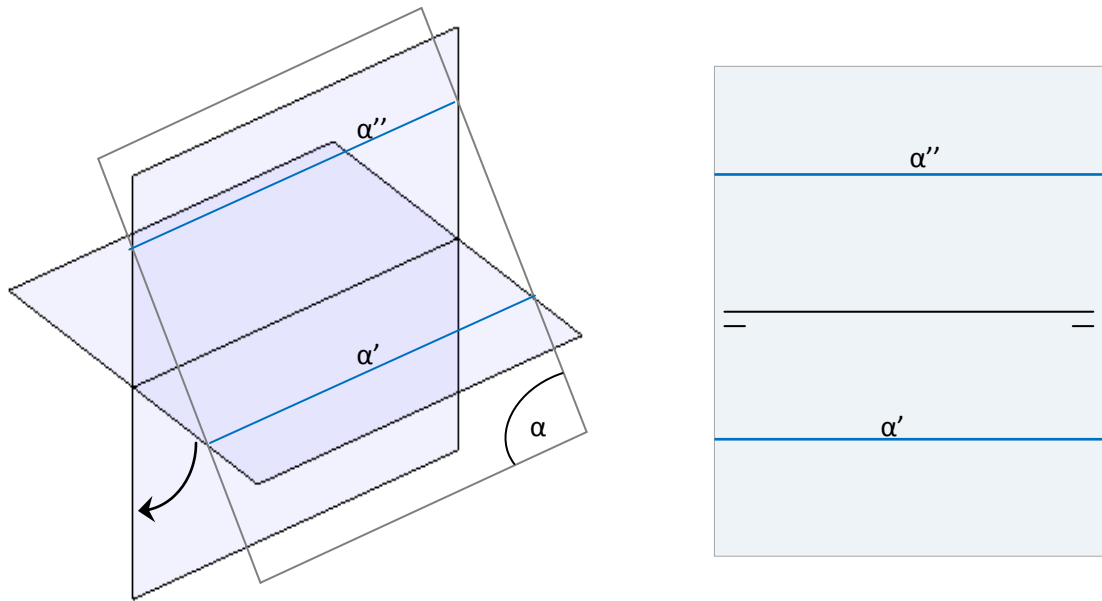


Fig. 4.25 Plano Paralelo a LT

El dibujo en 3D de un plano paralelo a LT de la Fig. 4.25 se observa una Y y una Z constante en la cualquiera de sus trazas, dando lugar a una recta paralela línea de tierra, con lo que uno de los vectores directores podría ser $\vec{u} = (u_x, 0, 0)$, y teniendo otro vector director $\vec{v} = (v_x, v_y, v_z)$, y el punto del plano $P_0 = (x_0, y_0, z_0)$ la ecuación general será:

$$\begin{vmatrix} x - x_0 & u_x & v_x \\ y - y_0 & 0 & v_y \\ z - z_0 & 0 & v_z \end{vmatrix} = (y - y_0)(u_x v_z) + (z - z_0)(u_x v_y) = b(y - y_0) + c(z - z_0)$$

$$\text{Siendo } a = 0 \rightarrow a = (u_y v_z - u_z v_y) \rightarrow u_y v_z = u_z v_y$$

$$by + cz + d = 0$$

$$\vec{u} = (u_x, u_y, \lambda u_y)$$

$$\vec{v} = (v_x, v_y, \lambda v_y)$$

16. Plano que pasa por LT

Como se muestra en la Fig. 4.26 un plano que pasa por Línea de Tierra es el plano que contiene a la Línea de Tierra. Sus dos trazas coinciden con Línea de Tierra, por lo que se recurre a una representación especial auxiliada por un punto del plano. Pasa por dos cuadrantes.

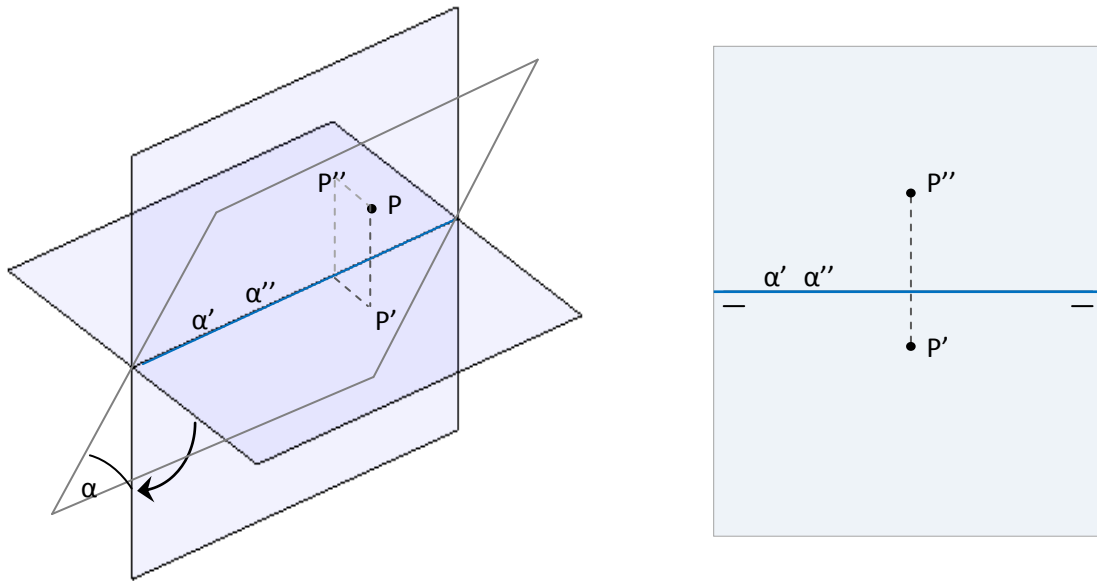


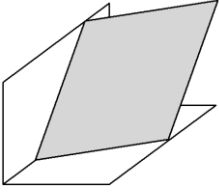
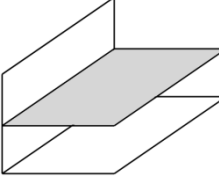
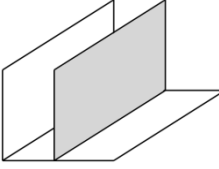
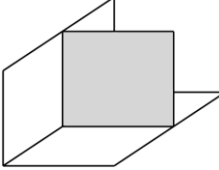
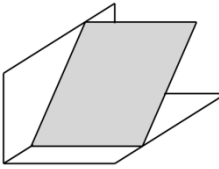
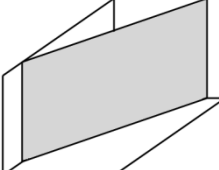
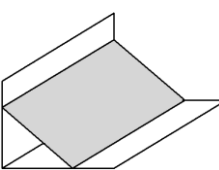
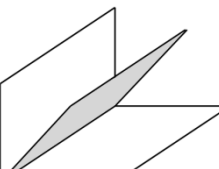
Fig. 4.26 Plano que pasa por LT

Un plano que pasa por la línea de tierra tiene las mismas características que un plano paralelo a la línea de tierra, con la peculiaridad de que pasa por el punto $P = (0,0,0)$, es decir:

$$\begin{vmatrix} x & u_x & v_x \\ y & 0 & v_y \\ z & 0 & v_z \end{vmatrix} = y(-u_x v_z) + z(u_x v_y) = by + cz$$

En la Tabla 4.3 se observa las relaciones que hay en los distintos tipos de planos así como sus ecuaciones, se muestra solo la parte visible, es decir el primer diedro.

Tabla 4.3 Planos vs Ecuaciones

TIPO PLANO	DIBUJO	RELACIONES	ECUACIONES
Oblicuo		Sin relación	$ax + by + cz + d = 0$ $\vec{u} = (u_x, u_y, u_z)$ $\vec{v} = (v_x, v_y, v_z)$
Horizontal		$z = cte$ $u_z = 0$ $v_z = 0$	$cz + d = 0$ $\vec{u} = (u_x, u_y, 0)$ $\vec{v} = (v_x, v_y, 0)$
Frontal		$y = cte$ $u_y = 0$ $v_y = 0$	$by + d = 0$ $\vec{u} = (u_x, 0, u_z)$ $\vec{v} = (v_x, 0, v_z)$
Perfil		$x = cte$ $u_x = 0$ $v_x = 0$	$ax + d = 0$ $\vec{u} = (0, u_y, u_z)$ $\vec{v} = (0, v_y, v_z)$
Canto		$\frac{v_x}{v_z} = \frac{u_x}{u_z}$	$ax + cz + d = 0$ $\vec{u} = (u_x, u_y, \lambda u_x)$ $\vec{v} = (v_x, v_y, \lambda v_x)$
Vertical		$\frac{v_x}{v_y} = \frac{u_x}{u_y}$	$ax + by + d = 0$ $\vec{u} = (u_x, \lambda u_x, u_z)$ $\vec{v} = (v_x, \lambda v_x, v_z)$
Paralelo a LT		$\frac{v_y}{v_z} = \frac{u_y}{u_z}$	$by + cz + d = 0$ $\vec{u} = (u_x, u_y, \lambda u_y)$ $\vec{v} = (v_x, v_y, \lambda v_y)$
Pasa por LT		$\frac{v_y}{v_z} = \frac{u_y}{u_z}$	$by + cz = 0$ $\vec{u} = (u_x, u_y, \lambda u_y)$ $\vec{v} = (v_x, v_y, \lambda v_y)$

5. MATLAB

5.1. Introducción al MATLAB

MATLAB es un lenguaje de computación técnica de alto nivel y un entorno interactivo para desarrollo de algoritmos, visualización de datos, análisis de datos y cálculo numérico. Con MATLAB, se pueden resolver problemas de cálculo técnico más rápidamente que con lenguajes de programación tradicionales. MATLAB es un programa de cálculo numérico orientado a matrices. Por tanto, será más eficiente si se diseñan los algoritmos en términos de matrices y vectores.

MATLAB se puede usar en una amplia gama de aplicaciones que incluyen procesamiento de señales e imágenes, comunicaciones, diseño de sistemas de control, sistemas de prueba y medición, modelado y análisis financiero y biología computacional. Los conjuntos de herramientas complementarios (colecciones de funciones de MATLAB para propósitos especiales, que están disponibles por separado) amplían el entorno de MATLAB permitiendo resolver problemas especiales en estas áreas de aplicación.

Además, MATLAB contiene una serie de funciones para documentar y compartir trabajo. Se puede integrar el código de MATLAB con otros lenguajes y aplicaciones, y distribuir los algoritmos y aplicaciones que se desarrollan usando MATLAB.

Las características principales del MATLAB son:

- Lenguaje de alto nivel para cálculo técnico
- Entorno de desarrollo para la gestión de código, archivos y datos
- Herramientas interactivas para exploración, diseño y resolución de problemas iterativos
- Funciones matemáticas para álgebra lineal, estadística, análisis de Fourier, filtraje, optimización e integración numérica
- Funciones gráficas bidimensionales y tridimensionales para visualización de datos
- Herramientas para crear interfaces gráficas de usuario personalizadas
- Funciones para integrar los algoritmos basados en MATLAB con aplicaciones y lenguajes externos, tales como C/C++, FORTRAN, Java, COM y Microsoft Excel.

5.1.1. Desarrollo de algoritmos y aplicaciones

El lenguaje MATLAB

El lenguaje MATLAB incluye operaciones vectoriales y matriciales que son fundamentales para resolver los problemas científicos y de ingeniería, agiliza tanto el desarrollo como la ejecución.

Con el lenguaje de MATLAB, se puede programar y desarrollar algoritmos más rápidamente que con los lenguajes tradicionales porque ya no es necesario realizar tareas administrativas de bajo nivel, tales como declarar variables, especificar tipos de datos y asignar memoria. En

muchos casos, MATLAB elimina la necesidad de bucles "for". En consecuencia, una línea de código de MATLAB generalmente reemplaza a varias líneas de código C o C++.

Al mismo tiempo, MATLAB ofrece todas las características de los lenguajes de programación tradicionales, que incluyen operadores aritméticos, control de flujo, estructuras de datos, tipos de datos, programación orientada a objetos (OOP) y depuración.

MATLAB permite ejecutar comandos o grupos de comandos uno a uno, sin compilar ni enlazar, y repetir su ejecución hasta lograr la solución óptima.

Para ejecutar rápidamente cálculos matriciales y vectoriales complejos, MATLAB utiliza bibliotecas optimizadas para el procesador. Para cálculos escalares de aplicación general, MATLAB genera instrucciones en código máquina utilizando su tecnología JIT (Just-In-Time).

Gracias a esta tecnología, que está disponible para la mayoría de las plataformas, las velocidades de ejecución son mucho más rápidas que las de los lenguajes de programación tradicionales.

Herramientas de desarrollo

MATLAB incluye herramientas de desarrollo que le ayudan a implementar sus algoritmos eficientemente. Las siguientes son algunas de ellas:

MATLAB Editor - Funciones de edición y depuración estándar, como establecimiento de puntos de interrupción y simulaciones paso a paso.

M-Lint Code Checker - Analiza el código y recomienda modificaciones para mejorar el rendimiento y mantenimiento.

MATLAB Profiler - Registra el tiempo que tarda en ejecutarse cada línea de código .

Directory Reports - Explora todos los archivos de un directorio y crea informes sobre la eficiencia del código, las diferencias entre los archivos, las dependencias de los archivos y la cobertura del código.

Diseño de interfaces gráficas de usuario

MATLAB dispone de la herramienta interactiva GUIDE (Graphical User Interface Development Environment) para diseñar y editar interfaces de usuario. Esta herramienta permite incluir listas de selección, menús desplegables, botones de pulsación, botones de opción y deslizadores, así como diagramas de MATLAB y controles ActiveX. También se pueden crear interfaces gráficas de usuario por medio de programación usando las funciones de MATLAB.

5.2. Interfaces Gráficas con MATLAB

MATLAB permite desarrollar de manera simple un conjunto de pantallas con botones, menús, ventanas, etc., que permiten utilizar de manera muy simple programas realizados en el entorno Windows. Este conjunto de herramientas se denomina interface de usuario.

Para poder hacer programas que utilicen las capacidades gráficas avanzadas de MATLAB hay que conocer algunos conceptos que se explican en los apartados siguientes, aunque, como se ha dicho, MATLAB dispone ahora de la herramienta GUIDE, que permite generar interfaces de usuario de una forma muy cómoda y sencilla.

5.2.1. Estructura de los gráficos de MATLAB

Los gráficos de MATLAB tienen una estructura jerárquica formada por objetos de distintos tipos. Esta jerarquía tiene forma de árbol, con el aspecto mostrado en la Fig. 5.1.

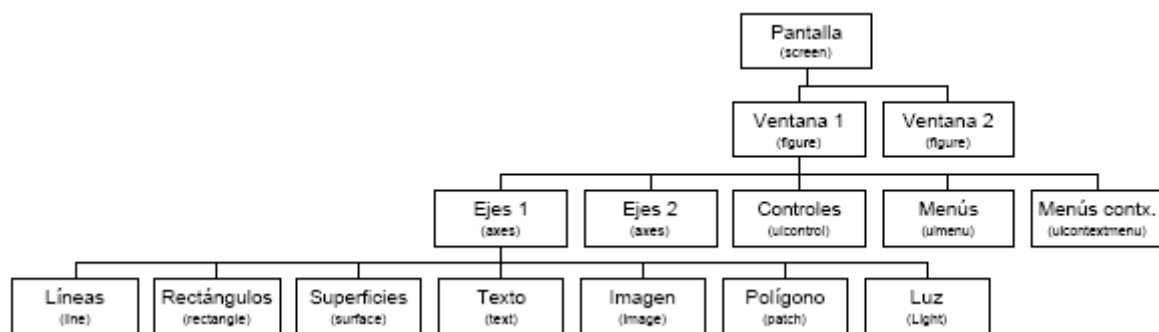


Fig. 5.1 Jerarquía Gráfica de MATLAB

Según se muestra en la Fig. 5.1 el objeto más general es la pantalla (screen). Este objeto es la raíz de todos los demás y sólo puede haber un objeto pantalla. Una pantalla puede contener una o más ventanas (figures). A su vez cada una de las ventanas puede tener uno o más ejes de coordenadas (axes) en los que representar otros objetos de más bajo nivel. Una ventana puede tener también controles (uicontrol) tales como botones, barras de desplazamiento, botones de selección o de opción, etc.) y menús (uimenu). Finalmente, los ejes pueden contener los seis tipos de elementos gráficos que permite MATLAB: líneas (line), rectángulos (rectangle), polígonos (patches), superficies (surface), imágenes bitmap (image) y texto (text) [4].

La jerarquía de objetos mostrada en la Fig. 5.1 indica que en MATLAB hay objetos padres e hijos. Por ejemplo, todos los objetos ventana son hijos de pantalla, y cada ventana es padre de los objetos ejes, controles o menús que están por debajo. A su vez los elementos gráficos (líneas, polígonos, etc.), son hijos de un objeto ejes, y no tienen otros objetos que sean sus hijos.

IDENTIFICADORES (HANDLES)

Cada uno de los objetos de MATLAB tiene un identificador único (handle). Algunos gráficos tienen muchos objetos, en cuyo caso tienen múltiples handles. El objeto raíz (pantalla) es siempre único y su identificador es el cero. El identificador de las ventanas es un entero, que aparece en la barra de nombre de dicha ventana. Los identificadores de otros elementos gráficos son números float, que pueden ser obtenidos como valor de retorno y almacenados en variables de MATLAB.

MATLAB puede tener varias ventanas abiertas, pero siempre hay una y sólo una que es la ventana activa. A su vez una ventana puede tener varios ejes (axes), pero sólo unos son los ejes activos. MATLAB dibuja en los ejes activos de la ventana activa. Los identificadores de la ventana activa, de los ejes activos y del objeto activo se pueden obtener respectivamente con los comandos:

gcf (*get current figure*) devuelve un entero, que es el handle de la ventana activa.

gca (*get current axes*) devuelve el handle de los ejes activos.

gco (*get current object*) devuelve el handle del objeto activo.

Estos valores de retorno pueden ser almacenados en variables para un uso posterior. En ocasiones el id es un vector de valores, como por ejemplo al crear una superficie que consta de líneas y polígonos. Los comandos anteriores han abierto una ventana y dibujado sobre ella dos líneas cruzadas de color amarillo y un triángulo de color verde.

En una interfaz gráfica sería el puntero a todas variables y elementos de la aplicación.

Todos los valores de las propiedades de los elementos y los valores de las variables transitorias del programa se guardan en una estructura (estructura matlab: puede almacenar matrices, string, arreglos, vectores, etc.) los cuales son accedidos mediante un único y mismo puntero para todos estos.

Propiedades de los objetos

Todos los objetos de MATLAB tienen distintas propiedades. Algunas de éstas son el tipo, el estilo, el padre, los hijos, si es visible o no, y otras propiedades particulares del objeto concreto de que se trate. Algunas de las propiedades comunes a todos los objetos son: children, clipping, parent, type, UserData, Visible y Tag. Otras propiedades son propias de un tipo determinado de objeto.

Las propiedades tienen valores por defecto, que se utilizan siempre que el usuario no indique otra cosa. Es posible cambiar las propiedades por defecto, y también devolverles su valor original (llamado factory, por ser el valor por defecto con que salen de fábrica). El usuario puede consultar (query) los valores de las propiedades de cualquier objeto. Algunas propiedades pueden ser modificadas y otras no (son read only). Hay propiedades que pueden tener cualquier valor y otras que sólo pueden tener un conjunto limitado de valores (por ejemplo, on y off).

FUNCIONES SET() Y GET()

MATLAB dispone de las funciones set y get para consultar y cambiar el valor de las propiedades de un objeto. Las funciones set(id) lista en pantalla todas las propiedades del objeto al que corresponde el handle (sólo los nombres, sin los valores de las propiedades). La función get(id) produce un listado de las propiedades y de sus valores.

Para conocer el valor de una propiedad particular de un objeto se utiliza la función *get(id,'propiedad')*.

Las propiedades de un objeto pueden ser cambiadas o modificadas (salvo que sean read-only) con el comando `set(id,'propiedad','valor')`. El comando `set` permite cambiar varias propiedades a la vez, poniendo sus nombres entre apóstrofes seguidos de sus valores.

Es posible también establecer las propiedades en el momento de la creación del objeto.

Se puede utilizar la propiedad `type` para saber qué tipo de objeto (línea, polígono, texto, ...) corresponde a un determinado id. Esto es especialmente útil cuando el id es un vector de valores correspondientes a objetos de distinto tipo.

Anteponiendo la palabra `Default` al nombre de un objeto y de una propiedad se puede acceder al valor por defecto de una propiedad, bien para consultar su valor, bien para modificarlo.

5.2.2. Creación de controles gráficos: Comando `uicontrol`

MATLAB permite desarrollar programas con el aspecto típico de las aplicaciones de Windows. Para todos los controles, se utilizará la función `uicontrol`, que permite desarrollar dichos controles. La forma general del comando `uicontrol` es la siguiente:

```
id_control = uicontrol(id_parent,'Propiedad1',valor1,'Propiedad2',valor2,(otras propiedades)
'callback','sentencias')
```

Las propiedades son las opciones del comando, que se explican en el apartado siguiente. Éstas tendrán comillas simples (') a su izquierda y derecha, e irán seguidas de los parámetros necesarios.

En caso de que el conjunto de propiedades de un control exceda una línea de código, es posible continuar en la línea siguiente, poniendo tres puntos seguidos (...).

El comando `uicontrol` permite definir los controles gráficos de MATLAB. Estos controles reciben las acciones de los usuarios, que se denominan eventos (por ejemplo, clicar en un botón, cambiar el valor de una barra de desplazamiento, ...).

Tipos de `uicontrol`

Existen ocho tipos de controles diferentes. La utilización de cada uno vendrá dada en función de sus características y aplicación.

- **PUSH BUTTONS**
Botón que al clicar sobre él con el ratón se producirá un evento que lanza una acción que deberá ser ejecutada por MATLAB.
- **TOGGLE BUTTONS**
Igual a los push button pero estos, permanecen activos hasta que se vuelve a clicar sobre ellos.
- **CHECK BOXES**
Permiten al usuario seleccionar entre dos opciones. Actúan como interruptores, indicando un estado on u off.
- **RADIO BUTTONS**

Permiten al usuario seleccionar entre varias posibilidades. La diferencia fundamental reside en que en los check boxes, las opciones son excluyentes, es decir, no puede haber más de uno activado, mientras que el control anterior permite tener una o más cajas activadas.

- **BARRAS DE DESPLAZAMIENTO**

Permiten al usuario introducir un valor entre un rango de valores. El usuario puede cambiar el valor clicando sobre la barra, licando en las flechas laterales o bien arrastrando el elemento central con el ratón.

- **CAJAS DE SELECCIÓN**

Permiten elegir una opción entre varias mostradas en una lista. Eligiendo una de las opciones, MATLAB realizará la opción elegida.

- **CAJAS DE TEXTO**

Son controles especiales, ya que no permite realizar ninguna operación con el ratón. Permiten escribir un texto en la pantalla.

- **CAJAS DE TEXTO EDITABLES**

Las cajas de texto se utilizan para introducir y modificar cadenas de caracteres. Puede tener una o más líneas, y la llamada a la opción de ejecución Callback será efectiva una vez que se pulsen las teclas Control-Return o se clique con el ratón fuera del control.

- **FRAMES**

Es un marco, su función es la de englobar una serie de opciones (botones, cajas de texto, etc....) con el fin de mantener una estructura ordenada de controles, separando unos de otros en función de las características del programa y del gusto del programador.

CALLBACK

Este comando especifica la acción a efectuar por MATLAB al actuar sobre el control. Se trata de una expresión u orden, almacenada en una cadena de caracteres o en una función, que se ejecutará al activar el control. Esta instrucción es equivalente a ejecutar la función o a realizar `eval('expresión')`.

Algunos controles tienen distintos tipos de callback según el evento que reciban del usuario.

La traducción de Callback es “devolver llamada”, en MATLAB se refiere las “funciones de llamada”, que se ejecutarán para las diferentes acciones realizadas en un input dentro de una interfaz gráfica, como por ejemplo hacer “click” en un botón llamará una función callback asociada a este evento.

Callback es una función que se escribe y se asocian con un componente específico en la GUI o interfaz gráfica de usuario con la figura misma. Las funciones callbacks controlan el GUI o el comportamiento de los componentes por realizar alguna acción en respuesta a un evento para sus componentes. El evento puede ser un clic del ratón sobre un botón, la selección de menú, presione la tecla, etc. Es decir es un tipo de programación orientada a eventos.

Las funciones callbacks proporcionan un control de cómo la interfaz gráfica responde a los eventos, tales como pulsaciones de los botones, el movimiento deslizante, la selección del

menú, o la creación y supresión de los componentes. Hay una serie de llamadas para cada componente y para la interfaz gráfica de la figura en sí.

Cuando ocurre un evento de un componente, el software MATLAB invoca la devolución de llamada componente que se asocia con ese evento. Como un ejemplo, supongamos que una interfaz gráfica de usuario tiene un botón que activa el trazado de algunos datos. Cuando el usuario pulsa el botón, el programa pide la devolución de llamada que asociados con hacer clic en ese botón y, a continuación, la devolución de llamada, que usted ha programado, obtiene los datos y argumentos que.

Un componente puede ser cualquier dispositivo de control, como uno ejes, botón, cuadro de lista, o control deslizante. Para efectos de la programación, también puede ser un menú, barra de herramientas, o un contenedor, como un panel o grupo de botones. Ver Componentes disponible para una lista y descripción de los componentes.

5.3. Guide

Matlab Guide es un entorno de programación visual que ofrece Matlab para poder realizar y ejecutar programas de Simulación a medida de forma simple, tiene las características básicas de todos los programas visuales como Visual Basic o Visual C++. Pcrear GUIs (Graphical User Interfaces) fácil y rápidamente dando auxiliando en el diseño y presentación de los controles de la interfaz, reduciendo la labor al grado de seleccionar, tirar, arrastrar y personalizar propiedades [4].

Una vez que los controles están en posición se editan las funciones de llamada (Callback) de cada uno de ellos, escribiendo el código de MATLAB que se ejecutará cuando el control sea utilizado. Siempre será difícil diseñar GUIs, pero no debería ser difícil implementarlas. GUIDE está diseñado para ser menos tediosos el proceso de aplicación de la interfaz grafica y obviamente para trabajar como herramienta de trazado de GUIs, entre sus poderosos componentes esta el editor de propiedades (property editor), este se encuentra disponible cualquier momento que se esté lidiando con los controles de MATLAB, el editor de propiedades por separado se puede concebir como una herramienta de trazado, y asistente de codificación (revisión de nombres y valores de propiedades). Cuando se fusiona con el panel de control, el editor de menú, y herramienta de alineación, resulta una combinación que brinda inigualable control de los gráficos en MATLAB.

Con una GUI, el flujo de cómputo está controlado por las acciones en la interfaz. Mientras que en un guión el flujo de comandos está predeterminado, el flujo de operaciones con una GUI no lo está. Los comandos para crear una interfaz con el usuario se escribe en un guión, la interfaz invoca el guión que se ejecute, mientras la interfaz del usuario permanece en la pantalla aunque no se haya completado la ejecución del guión.

En la Fig. 5.2 se muestra el concepto básico de la operación del software con una GUI. Cuando se interactúa con un control, el programa registra el valor de esa opción y ejecuta los comandos prescritas en la cadena de invocación. Los menús de interfaz con el usuario, los botones, los menús desplegables, los controladores deslizantes y el texto editable son

dispositivos que controlan las operaciones del software. Al completarse la ejecución de las instrucciones de la cadena de invocación, el control vuelve a la interfaz para que puedan elegirse otra opción del menú. Este ciclo se repite hasta que se cierra la GUI.

El control guarda un string que describe la acción a realizar cuando se invoca puede consistir en un solo comando de MATLAB o una secuencia de comandos, o en una llamada a una función. Es recomendable utilizar llamadas a funciones, sobre todo cuando se requieren de más de unos cuantos comandos en la invocación.

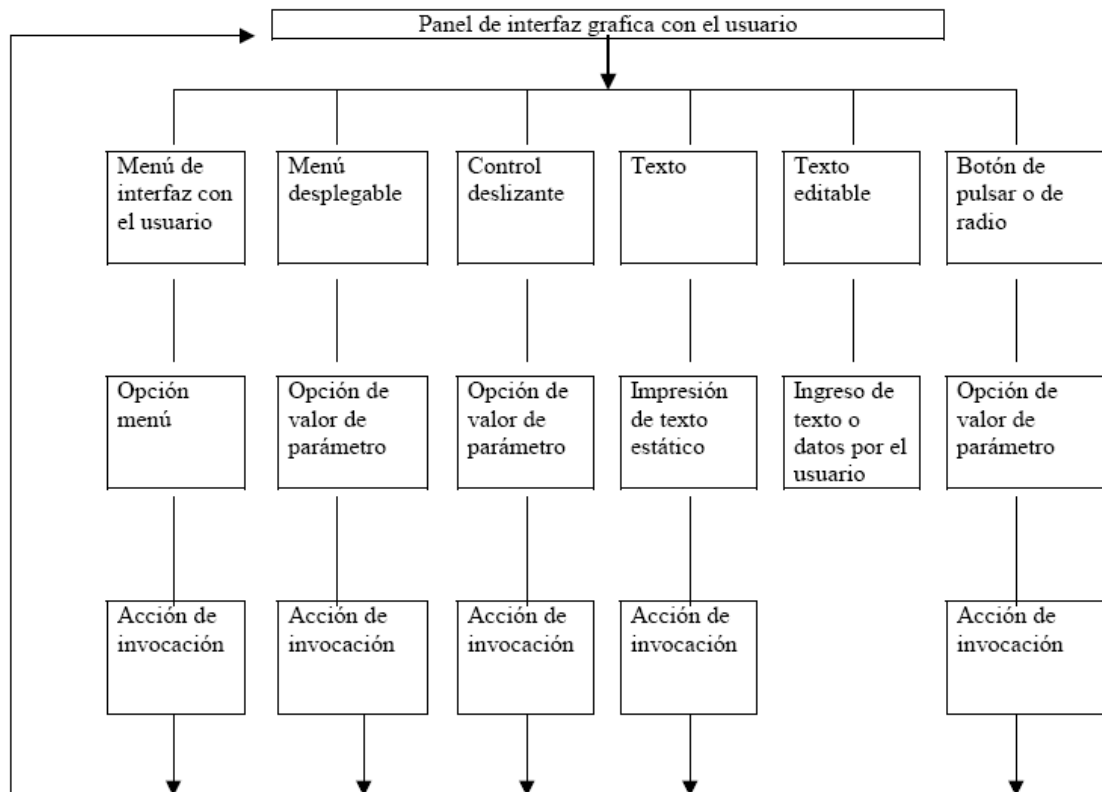


Fig. 5.2 Operación Software del GUI

Un guide consta de dos archivos uno “.m” (ejecutable) y otro “.fig” la parte grafica. Las dos partes están unidas a través de las subrutinas callback.

TÍTULO III: SISTEMA DIÉDRICO CON MATLAB

6. Elecciones Previas

6.1. Funcionalidad

Esta aplicación es un programa que, con el MATLAB como software, representa elementos que el usuario introduce en dos y tres dimensiones, utilizando los fundamentos del Sistema Diédrico explicados anteriormente, los dos dibujos se pueden ver en la misma pantalla uno al lado del otro de forma que se puedan comparar y así aclarar los conceptos.

Se busca el uso sencillo y dinámico de la herramienta y, sobre todo, se pretende la claridad del dibujo para el esfuerzo del usuario para la visualización de los elementos representados.

En aras a un mayor dinamismo se permite una sencilla selección de elementos, minimizando la necesidad de utilización del teclado para introducir datos.

El usuario puede interactuar para una visualización clara de los elementos, ajustando la representación a sus conveniencias, ya sea seleccionando los elementos que le interesa mostrar, eliminando las líneas de menos información, seleccionando los nombres a mostrar o abriendo figuras independientes de cada uno de los dibujos, pudiendo ajustarlas al tamaño deseado y, en el caso de la figura en tres dimensiones, girar el cubo que engloba el dibujo.

6.2. Interfaz Gráfica

Para un control más absoluto de la Interfaz gráfica y debido a las limitaciones de la herramienta de diseño de GUI's de MATLAB (GUIDE), entre ellas la imposibilidad de dibujar planos con transparencia en tres dimensiones (usando la función `surf`), se ha optado por su codificación directa, creando desde un fichero la figura con todos sus elementos y dándole los atributos, eventos... que sean necesarios.

La Interfaz Gráfica implementada tendrá la limitación de la configuración de pantalla. Está preparada para una configuración de 1280 por 1024 píxeles. Si la configuración es demasiado pequeña, puede no entrar en la pantalla del usuario.

6.3. Geometría

Esta aplicación representa los puntos a través de las coordenadas introducidas directamente por el usuario y, a partir de ellos los demás elementos. Con dos puntos se crea una recta. En el caso de los planos hay más posibilidades: tres puntos, un punto y una recta o dos rectas. Para guardar los elementos que se van dibujando se ha requerido la selección de las características principales de cada tipo de elemento, eligiendo entre las posibilidades expuestas en el capítulo El Sistema Diédrico, las siguientes:

Punto: Para cada punto se guardan sus tres coordenadas:

$$(x, y, z)$$

Recta: Para cada recta se guarda un vector director y un punto perteneciente a la recta.

La recta se crea a partir de dos puntos $P_1 = (x_1, y_1, z_1)$ y $P_2 = (x_2, y_2, z_2)$, con los que se monta el vector a guardar. El vector director será la diferencia entre ellos:

$$\vec{v} = (v_x, v_y, v_z) = \overrightarrow{P_1 P_2} = P_2 - P_1 = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

En resumen, se guarda el vector:

$$(x_1, y_1, z_1, v_x, v_y, v_z) = (x_1, y_1, z_1, x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

Plano: Por cada plano se guarda su ecuación cartesiana.

El plano se puede crear a partir de rectas y/o puntos, para calcular la ecuación cartesiana se utilizan tres puntos (en el caso de las rectas se parte de los puntos a partir de los cuales se creó) $P_1 = (x_1, y_1, z_1)$, $P_2 = (x_2, y_2, z_2)$ y $P_3 = (x_3, y_3, z_3)$ para obtener la ecuación cartesiana $\pi: ax + by + cz + d$, que se calcula a partir de la igualdad:

$$\begin{vmatrix} x - x_0 & u_x & v_x \\ y - y_0 & u_y & v_y \\ z - z_0 & u_z & v_z \end{vmatrix} = 0, \text{ que despejando queda:}$$

$$a = (u_y v_z - u_z v_y) \quad b = (u_z v_x - u_x v_z) \quad c = (u_x v_y - u_y v_x) \quad d = -ax_0 - by_0 - cz_0$$

Siendo $\vec{u} = (u_x, u_y, u_z)$ y $\vec{v} = (v_x, v_y, v_z)$ los vectores directores que se calculan a partir de los puntos dados:

$$P_0 = (x_0, y_0, z_0) = P_1 = (x_1, y_1, z_1)$$

$$\vec{u} = (u_x, u_y, u_z) = \overrightarrow{P_1 P_2} = P_2 - P_1 = (x_2 - x_1, y_2 - y_1, z_2 - z_1)$$

$$\vec{v} = (v_x, v_y, v_z) = \overrightarrow{P_1 P_3} = P_3 - P_1 = (x_3 - x_1, y_3 - y_1, z_3 - z_1)$$

Con lo que queda:

$$a = ((y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1))$$

$$b = ((z_2 - z_1)(x_3 - x_1) - (x_2 - x_1)(z_3 - z_1))$$

$$c = ((x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1))$$

$$d = -ax_1 - by_1 - cz_1$$

En resumen, se guarda el vector:

$$(a, b, c, d)$$

6.4. Nomenclatura y elementos

Para la representación de los elementos hacen falta un conjunto de proyecciones, trazas y líneas auxiliares. Cada uno de los elementos y cada uno de sus componentes se identifica con un texto que debe seguir una lógica.

En esta aplicación se han dividido en dos todos los componentes de cada tipo de elemento, unos serán los imprescindibles para la representación y visión del elemento, y otros los auxiliares, que se representarán por medio de líneas auxiliares, de tal forma que al ejecutar el programa se podrá seleccionar si pintar o no estas líneas auxiliares.

El Punto

Los puntos se representan con la letra "P" y un índice numérico sucesivo que se le da automáticamente al ser creado, así el primer punto será P1 y el cuarto P4. La representación del punto en el Sistema Diédrico se realiza a través de sus dos proyecciones en los planos coordenados, la proyección en el Plano Vertical se representa con doble comilla ("'), y en el Plano Horizontal con comilla simple ('). En la Fig. 6.1 se observa la representación del punto P1, donde la proyección vertical será P1'' y la proyección horizontal P1'.

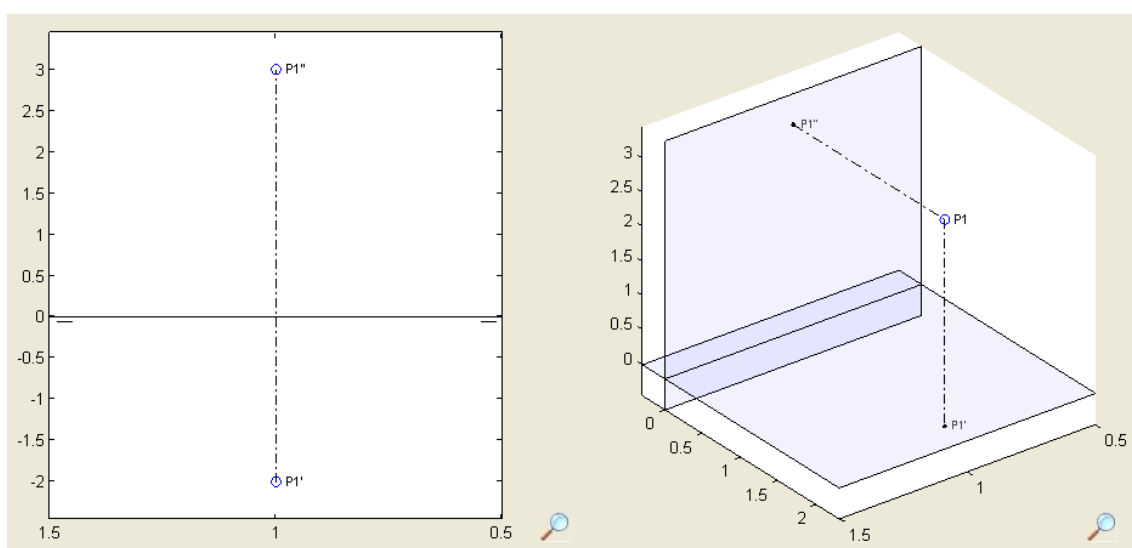


Fig. 6.1 Nomenclatura del Punto

En la representación del punto P1 en la Fig. 6.1 se ven las líneas auxiliares. El dibujo del punto sin líneas auxiliares en dos dimensiones será igual que con líneas auxiliares, es decir, como en la Fig. 6.1, con sus dos proyecciones identificadas con sus nombres y la línea que une éstas perpendicular a la Línea de Tierra. El punto sin líneas auxiliares en tres dimensiones será distinto, sólo se dibuja el punto con su nombre, al visualizar las líneas auxiliares se añaden las dos proyecciones con sus nombres y las líneas que las unen con el punto, como en la Fig. 6.1.

La Recta

Las rectas se nombran con la letra "r" y un índice numérico sucesivo que se les da automáticamente al ser creadas, así la primera recta será r1 y la tercera r3. La representación

de la recta en el Sistema Diédrico se obtiene a través de las dos proyecciones de la recta en los planos coordenados. La proyección en el Plano Vertical se representará con doble comilla ("), y en el Plano Horizontal con comilla simple ('). En la Fig. 6.2 se observa la recta $r1$ donde la proyección vertical será $r1''$ y la proyección horizontal $r1'$, además también tiene representadas sus trazas, con mismo índice que la recta; la traza vertical con la letra V y el índice de la recta y la traza horizontal con la letra H y el índice. Estas trazas se representan igual que cualquier otro punto, así, como se ve en la Fig. 6.2, la traza vertical de la recta $r1$ será $V1$ y su proyección vertical y horizontal son $V1''$ y $V1'$, y la traza horizontal será $H1$ y sus proyecciones vertical y horizontal son $H1''$ y $H1'$.

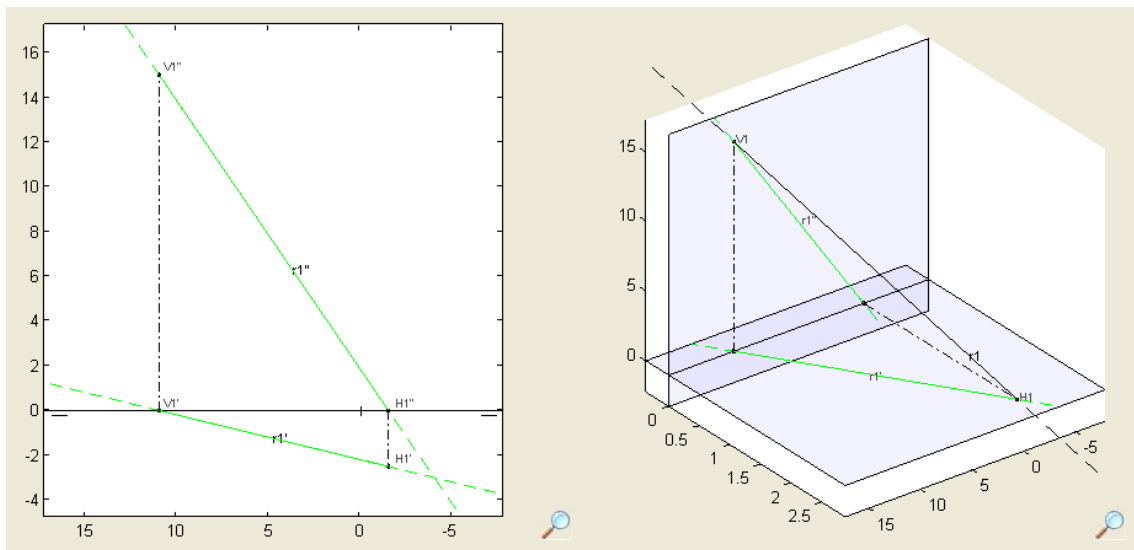


Fig. 6.2 Nomenclatura de la Recta

La recta de la Fig. 6.2 se ve representada con las líneas auxiliares, conteniendo mucho más detalle que sin líneas auxiliares. En el dibujo de la recta sin líneas auxiliares en dos dimensiones sólo se representan sus dos proyecciones, así como los nombres de éstas y un punto en cada una de las proyecciones de sus trazas, sin un texto identificativo, y en tres dimensiones se dibuja la recta con su nombre y sus trazas sin texto identificativo. El dibujo con líneas auxiliares como el que se ve en la Fig. 6.2 tiene los mismos elementos que sin ellas y, además, en dos dimensiones el nombre de las proyecciones de sus trazas y la línea perpendicular a la Línea de Tierra, que une las proyecciones. Y para tres dimensiones se añaden las dos proyecciones con su identificación, así como el nombre de las trazas y la línea que las une, perpendicular a la Línea de Tierra.

El Plano

Los planos se nombran con la letra "a" y un índice numérico sucesivo que se les da automáticamente al ser creados, así el primer plano será $a1$ y la tercero $a3$. La representación del plano en el Sistema Diédrico la forman las dos trazas del plano, la traza en el Plano Vertical se representará con doble comilla ("), y en el Plano Horizontal con comilla simple ('). En la Fig. 6.3 se observa la representación del plano $a1$ donde la traza vertical es $a1''$ y la traza horizontal del plano $a1$ será $a1'$.

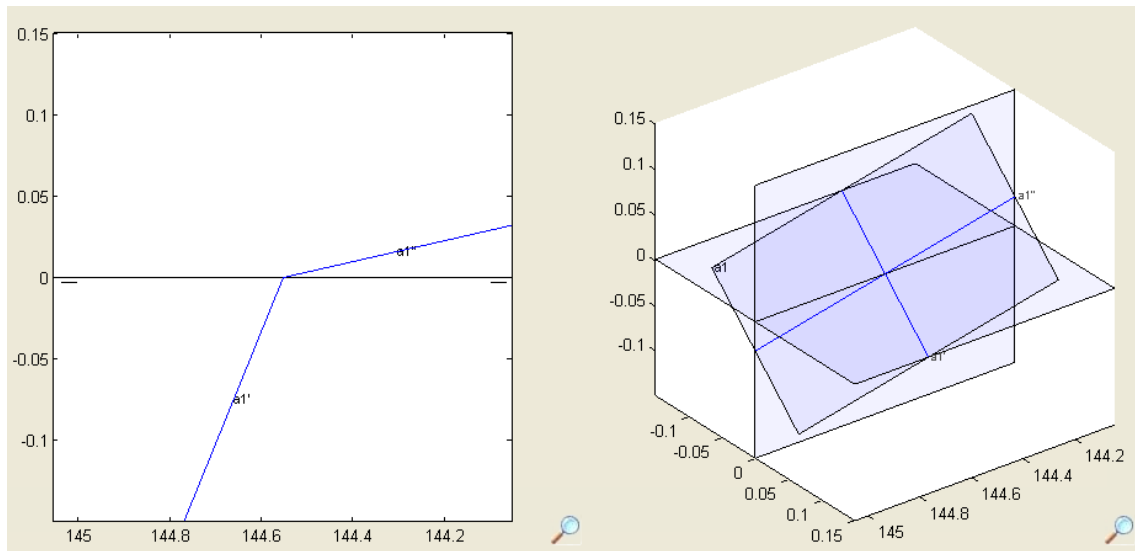


Fig. 6.3 Nomenclatura del Plano

El plano $a1$ representado en la Fig. 6.3 incluye las líneas auxiliares. En dos dimensiones se dibujan sus trazas y en tres dimensiones además de las trazas se dibuja el plano, en ambos casos con textos identificativos en cada uno de los componentes del dibujo. Un plano representado sin líneas auxiliares, sólo se diferencia con el representado con líneas auxiliares, en los textos identificativos de las trazas en tres dimensiones, que no aparecen.

7. Diseño Técnico

7.1. Diseño de Interfaz Gráfica

Al ejecutar el programa se abre una figura en la que se crean los componentes del programa (en MATLAB los *uicontrol*), seguidamente se llama a una función que limpia todos los componentes y dibuja el espacio de trabajo, quedando como se muestra en la Fig. 7.1.



Fig. 7.1 Inicialización del Espacio de Trabajo

Dentro de los *uicontrol* que se crean podrían distinguirse “activos” y “pasivos”, estos últimos son textos aclarativos. Los activos pueden ser botones, que lanzan eventos para llamar a las funciones, o elementos en los que el usuario introduce la información necesaria para ejecutar el programa.

La pantalla se divide en 4 zonas diferenciadas para su mejor organización y el mantenimiento del orden dentro lo posible al cambiar de configuración.

La zona de ACTUALIZACIÓN es la que aparece en la Fig. 7.2, en ella están todos los elementos que influyen en creación/borrado y dibujo de los elementos.

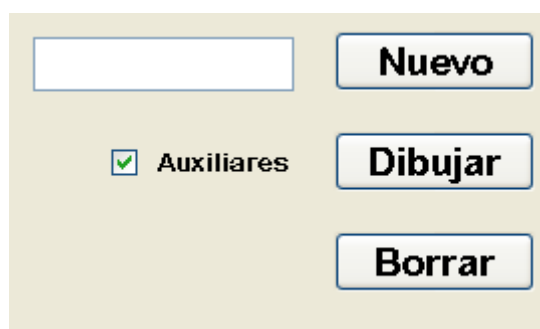


Fig. 7.2 Zona de Actualización

Nuevo: Botón con el literal “Nuevo” que sirve para crear elementos nuevos. Al presionarlo llama a la función “Func_Nuevo” (en este documento se le denomina “Nuevo”).

Borrar: Botón con el literal “Borrar” que sirve para borrar el elemento seleccionado. Al presionarlo llama a la función “Func_Borrar” (en este documento se le denomina “Borrar”).

Dibujar: Botón con el literal “Dibujar” que sirve para dibujar elementos ya creados. Al presionarlo llama a la función “Func_Dibujar” (en este documento se le denomina “Dibujar”).

elemento: Campo de texto donde se introduce los elementos a crear, para crear un punto es necesario, pero para una recta y un plano es opcional.

auxil: Checkbox con el texto “Auxiliares” que permite al usuario elegir si quiere dibujar con líneas auxiliares o sin ellas.

La zona de MEMORIA se muestra en la Fig. 7.3, en ella aparecen los elementos creados, actuando como memoria visible además de permitir su selección, ya sea para dibujar, borrar o crear otros elementos o para eliminar o añadir textos de elementos.

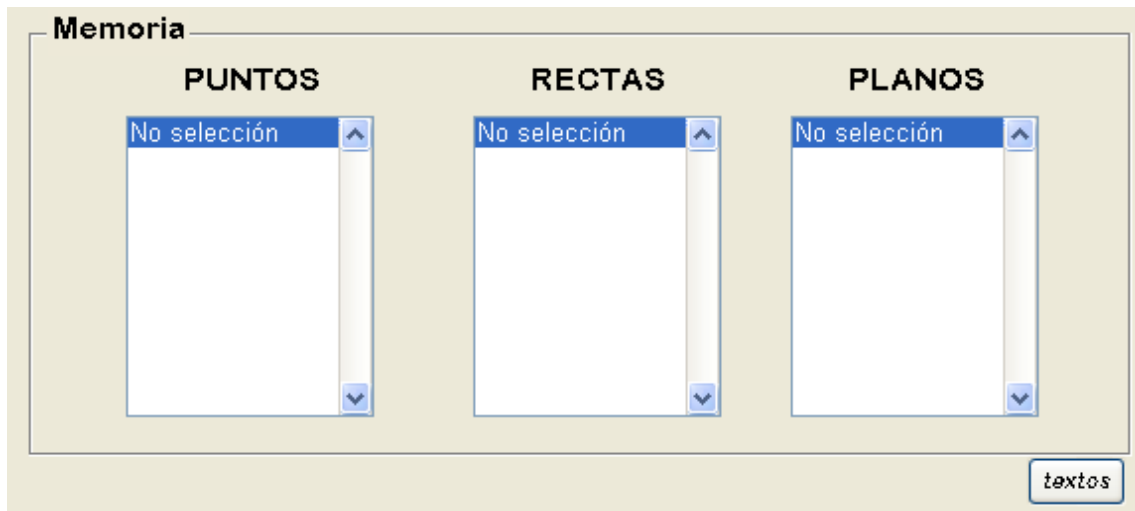


Fig. 7.3 Zona de Memoria

listpuntos: Listbox que contiene los nombres y coordenadas de los puntos. Permite al usuario hacer selecciones, controlar los textos y reconocer los puntos, a los que el mismo programa da nombre. Al hacer doble click en uno de sus elementos llama a la función Dibujar, y si es uno de los elementos ya representados elimina el texto si lo tiene o lo coloca si no lo tiene.

listrectas: Listbox que contiene los puntos a partir de los cuales se forma la recta. Permite al usuario hacer selecciones, controlar los textos y reconocer las rectas, a las que el mismo programa da nombre. Al hacer doble click en uno de sus elementos llama a

la función Dibujar, y si es uno de los elementos ya representados elimina el texto si lo tiene o lo coloca si no lo tiene.

listplanos: Listbox que contiene los puntos y/o rectas que forman. Permite al usuario hacer selecciones, controlar los textos y reconocer los planos, a los que el mismo programa da nombre. Al hacer doble click en uno de sus elementos llama a la función Dibujar, y si es uno de los elementos ya representados elimina el texto si lo tiene o lo coloca si no lo tiene.

textos: Botón con el literal “textos” que sirve para quitar o colocar todos los textos de los elementos dibujados. Al presionarlo llama a la función Dibujar.

La zona de DIBUJO se muestra en la Fig. 7.4, en ella se deja el espacio para los dibujos y aparecen los elementos para maximizar la representación.

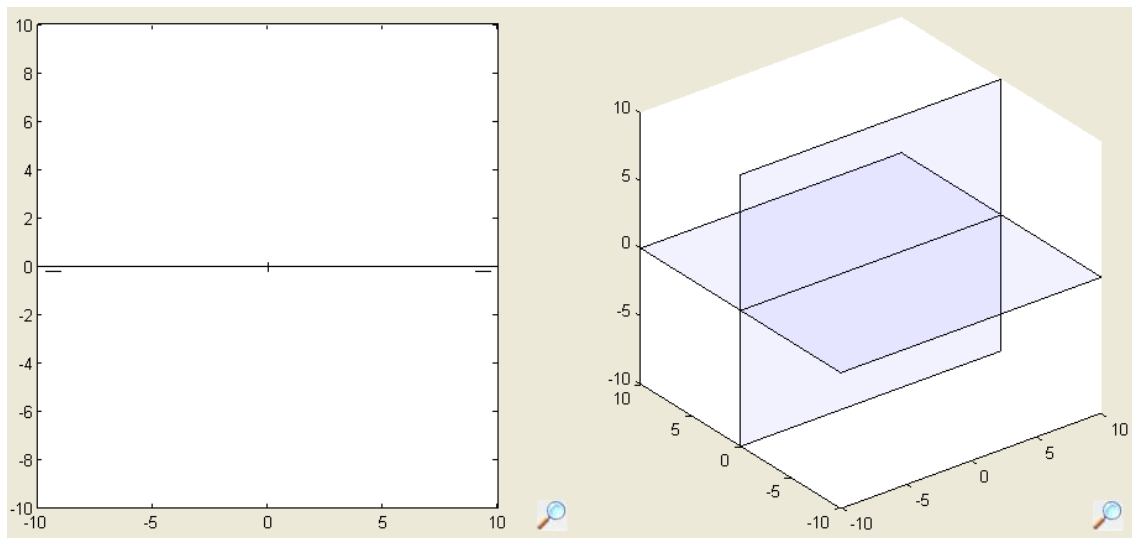


Fig. 7.4 Zona de Dibujo

Max2D: Botón con la imagen de una lupa, que sirve para agrandar en otra figura el mismo dibujo de dos dimensiones. Al presionarlo llama a la función Dibujar.

Max3D: Botón con la imagen de una lupa, que sirve para agrandar en otra figura el mismo dibujo de tres dimensiones. Al presionarlo llama a la función Dibujar.

La zona del PIE se muestra en la Fig. 7.5.

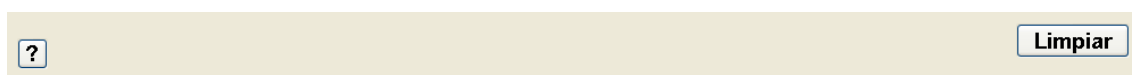


Fig. 7.5 Zona de Pie

Limpiar: Botón con el literal “Limpiar” que sirve para reiniciar el espacio de trabajo. Al presionarlo llama a la función “Func_Limpiar” (en este documento se le hará referencia con el nombre “Limpiar”).

AYUDA: Botón con el literal "?", que abre una figura nueva con un texto que explica el funcionamiento del programa, al presionarlo llama a la función "Figura_Ayuda" (en este documento se le denomina "Ayuda").

Elementos OCULTOS:

puntos: Objeto oculto que almacena las coordenadas de los puntos al ir creándose en la función Nuevo, guardando en el objeto un vector de dimensión $3 \times (\text{número de puntos})$, ya que cada punto ocupa tres posiciones, una por cada una de sus coordenadas.

rectas: Objeto oculto que almacena las características que determinan la recta al ir creándose en la función Nuevo, guardando en el objeto un vector de dimensión $6 \times (\text{número de puntos})$, ya que cada recta ocupa seis posiciones, tres para las coordenadas del vector director y otras tres para las de un punto perteneciente a la recta.

planos: Objeto oculto que almacena los factores de la ecuación del plano en la función Nuevo guardando en el objeto un vector de dimensión $4 \times (\text{número de puntos})$, ya que la ecuación del plano será $Ax + By + Cz + D = 0$, y cada plano ocupara cuatro posiciones para (A, B, C, D).

antele: Objeto oculto, que en la función Dibujar guarda los datos de las condiciones del dibujo, para poder repetir un dibujo exacto en caso de presionar alguna de las lupas.

Para pasar los componentes necesarios al llamar a las funciones se crea una estructura o puntero, de tal forma que el cambio de propiedades, recogida o cambio de los valores... se haga a través de ésta. Los componentes que se guardan son: el campo de texto "elemento", el checkbox "auxil", los listboxs, o memorias visibles "listpuntos", "listrectas" y "listplanos", las memorias ocultas "puntos", "rectas", "planos" y "antele" y la figura entera del espacio de trabajo (para poder identificar el doble click en los listbox). En resumen se guardan todos los componentes cuyo valor sea necesario recoger o modificar en algún punto del programa, que serán todos los *uicontrol* "activos" del programa, que no sean botones cuya única finalidad es lanzar eventos.

En cada uno de los *uicontrol* "activos", de la estructura mencionada, se guarda una información, ya sea en el *value* o en el *String* que forman parte de las propiedades del *uicontrol*. En el caso de los listbox contienen doble información, por una parte el *value* informa de la selección que tiene en ese momento, y por otra parte en la *String* es la memoria visible para el usuario, es decir, contiene cada uno de los elementos para que el usuario pueda identificarlos.

Los listbox son elementos "multitarea", ya que sirven de memoria visible para el usuario, al que además le permite reconocer los elementos por un nombre, permite la selección de los elementos y lanza eventos al presionar en uno de ellos.

7.2. Interacción con el Usuario

Esta aplicación tiene ocho botones, el usuario al presionar cada uno de ellos lanza un evento que llama a una función, así como al hacer doble click en uno de los elementos de cualquiera de los listbox, de esta forma decide la acción a realizar.

En el diagrama de la Fig. 7.6 se muestran todos los eventos que el usuario puede lanzar, cada uno de éstos llama a una función. Hay cuatro funciones fundamentales que engloban toda la actuación sobre los elementos: una función se encarga de reiniciar el espacio de trabajo (función Limpiar), otra de crear elementos (función Nuevo), otra de borrar elementos (función Borrar) y la principal que se encarga de dibujar (función Dibujar). Las funciones se llaman unas a otras en caso de ser necesario, pero ya no como un evento. Además de las principales, el usuario podrá llamar a una función de Ayuda para mostrar una figura que explica todas las opciones de la herramienta.

La función Nuevo se encarga de crear los elementos y guardarlos en la memoria visible (listbox de cada tipo de elemento) y en la memoria oculta (componentes). Se llama solamente desde el botón Nuevo con los datos necesarios introducidos y llama a la función Dibujar para que dibuje todos los elementos creados.

La función Borrar se encarga de borrar el elemento seleccionado, tanto de la memoria visible como de la memoria oculta, además de renombrar todos los elementos necesarios. (A esta función se la llama solamente desde el botón Borrar con el elemento a borrar seleccionado y llama a la función Dibujar para dibujar todos los elementos que se mantienen, en el caso de que no queden elementos por dibujar llama a la función Limpiar.

La función Limpiar se encarga de reiniciar el espacio de trabajo, tanto los componentes como los dibujos. Se llama desde el botón Limpiar, y se ejecuta desde la función Borrar si no hay elementos a dibujar y al abrir el programa.

La función Dibujar se encarga de hacer el dibujo, se lanza directamente al presionar el botón Dibujar, para dibujar los elementos seleccionados, con o sin líneas auxiliares, desde los botones Max2D y Max3D que dibuja en grande exactamente el mismo dibujo, ya sea en dos o tres dimensiones. También se llama a la función para actuar sobre los nombres que lo identifican, desde el botón textos para eliminar o colocar los nombres de todos los elementos representados y desde los listbox haciendo doble click en un elemento para que si está entre los dibujados, quite o coloque sus textos identificativos. También se ejecuta al final de la función Nuevo y la función Borrar, cuando quedan elementos por dibujar. Al tener tantas posibilidades, que en parte dependen del lugar desde donde ha sido llamado, tendrá más variables de entradas y más validaciones que las demás funciones.

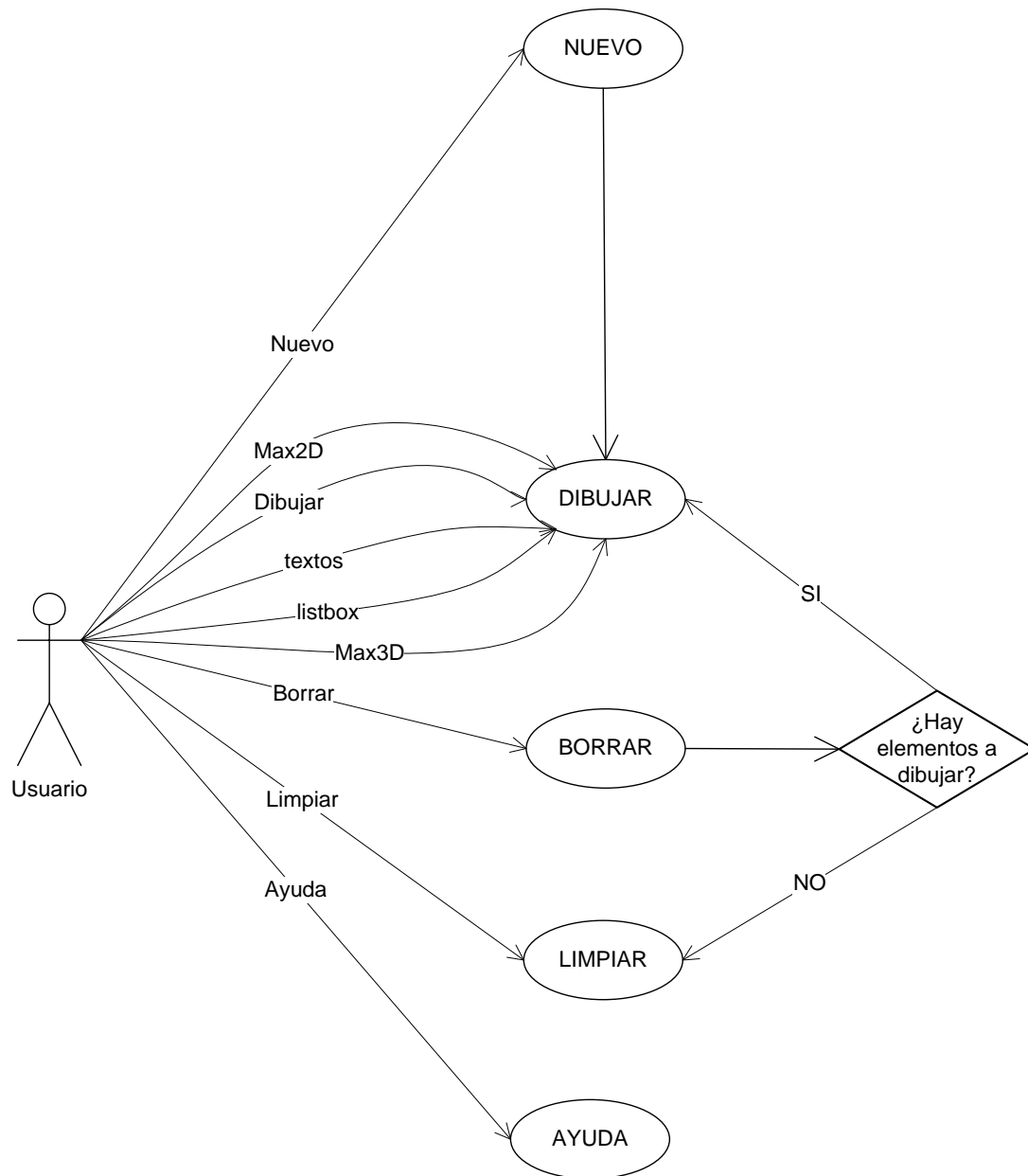


Fig. 7.6 Interacción con el Usuario

En resumen, como aparece representado en la Fig. 7.6 el usuario podrá lanzar los siguientes eventos:

- Presionando el botón “Ayuda” llama a la función Ayuda que muestra una figura con las explicaciones para manejar el programa.
- Desde los botones “Dibujar”, “Max2D”, “Max3D”, “textos” y el doble click en cualquiera de los elementos de algún listBox llama a la función Dibujar cada uno de ellos con unos parámetros de entradas distintos que condicionarán su ejecución.
- El botón “Limpiar” llamará a la función Limpiar.
- El botón “Borrar” llamará a la función Borrar, esta llama a su vez a la función Dibujar o Limpiar.
- El botón “Nuevo” llama a la función Nuevo, que a su vez llamará a la función Dibujar.

7.3. Implementación

Para actuar sobre los elementos hay que tener en cuenta que se trabaja con una memoria visible y otra oculta por cada tipo de elemento, éstas deben tener una relación entre ellas y con el elemento al que hacen referencia. Las memorias visibles son los elementos que el usuario ve en los listbox y la memoria oculta consiste en *uicontrols* ocultos que guardan vectores numéricos con las características de cada elemento.

Para las memorias visibles, la relación con el elemento es sencilla, en el listbox los elementos se cargan en una matriz, su primera posición no corresponde a ningún elemento, es el seleccionado por defecto y tiene el texto “Sin selección”, así que el primer elemento dibujado se guarda en la posición 2, el segundo en la 3 y así sucesivamente. Por ejemplo para la recta r5, se selecciona en el listbox de recta la posición 5+1. Siempre el elemento i se referencia con la posición $i+1$ del listbox correspondiente.

La relación de cada memoria oculta con los elementos depende del tipo de elemento:

- Del punto se guardan tres coordenadas en la memoria oculta “*puntos*”, por lo que cada elemento ocupa tres posiciones del vector, correspondiendo las tres primeras posiciones de “*puntos*” con el punto P1, las tres siguientes con el punto P2, etc. Llegando a que para un punto i se tienen las posiciones del vector que van desde $3(i - 1) + 1$ hasta $3(i - 1) + 3$.
- En la memoria oculta de las rectas “*rectas*” se guardan por cada una seis posiciones, correspondiendo las seis primeras posiciones de “*rectas*” con la recta r1, las seis siguientes con la recta r2, etc. Llegando a que para una recta i se tienen las posiciones del vector que van de $6(i - 1) + 1$, hasta $6(i - 1) + 6$.
- En la memoria oculta de los planos “*planos*” se guardan por cada uno cuatro posiciones, correspondiendo las cuatro primeras posiciones de “*rectas*” con el plano a1, las cuatro siguientes con el plano a2, etc. Llegando a que para un plano i se tienen las posiciones del vector que van de $4(i - 1) + 1$, hasta $4(i - 1) + 4$.

7.3.1. Inicializar Espacio de Trabajo: Función LIMPIAR

La función Limpiar, se encarga de reiniciar el espacio de trabajo, como parámetro de entrada sólo tiene la estructura o puntero que se ha formado al iniciar el programa de los componentes que se modifican.

En la Fig. 7.7 se ve la secuencia general que sigue la función Limpiar:

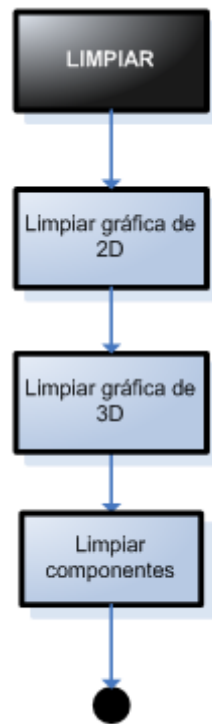


Fig. 7.7 Diagrama de Limpiar

Para la gráfica en dos dimensiones se dibujará solamente la Línea de Tierra, que como aparece en la Fig. 7.8, se representa con una línea horizontal que va de extremo a extremo del dibujo y un trazo corto en cada uno de los extremos. En tres dimensiones se representan los planos de proyección y la línea donde cortan, que corresponde con la Línea de Tierra.



Fig. 7.8 Espacio de Trabajo Inicial

Los componentes se limpian como se muestra en la Fig. 7.8, los listbox se vaciarán, dejando solamente una posición con el texto “Sin selección”, el campo de texto se vacía y el checkbox se deja seleccionado. Y los componentes ocultos se vaciarán.

7.3.2. Creación de Elementos: Función NUEVO

La función Nuevo, se encarga de crear y guardar los elementos. Como parámetro de entrada sólo tiene la estructura o puntero que se ha formado al iniciar el programa de los componentes que se modifican. Al final la función llama a Dibujar.

A la función Dibujar se le manda lo necesario para hacer la representación de todos los dibujos sin líneas auxiliares, y con los nombres identificativos.

En esta función se recogen datos introducidos por el usuario, como son las coordenadas para los puntos, o los elementos que forman las rectas o los planos, estos datos pueden estar mal introducidos produciendo los llamados “errores de inserción” que son controlados por el programa y se informan en forma de avisos al usuario.

En la Fig. 7.9 se ve la secuencia general que sigue la función Nuevo:

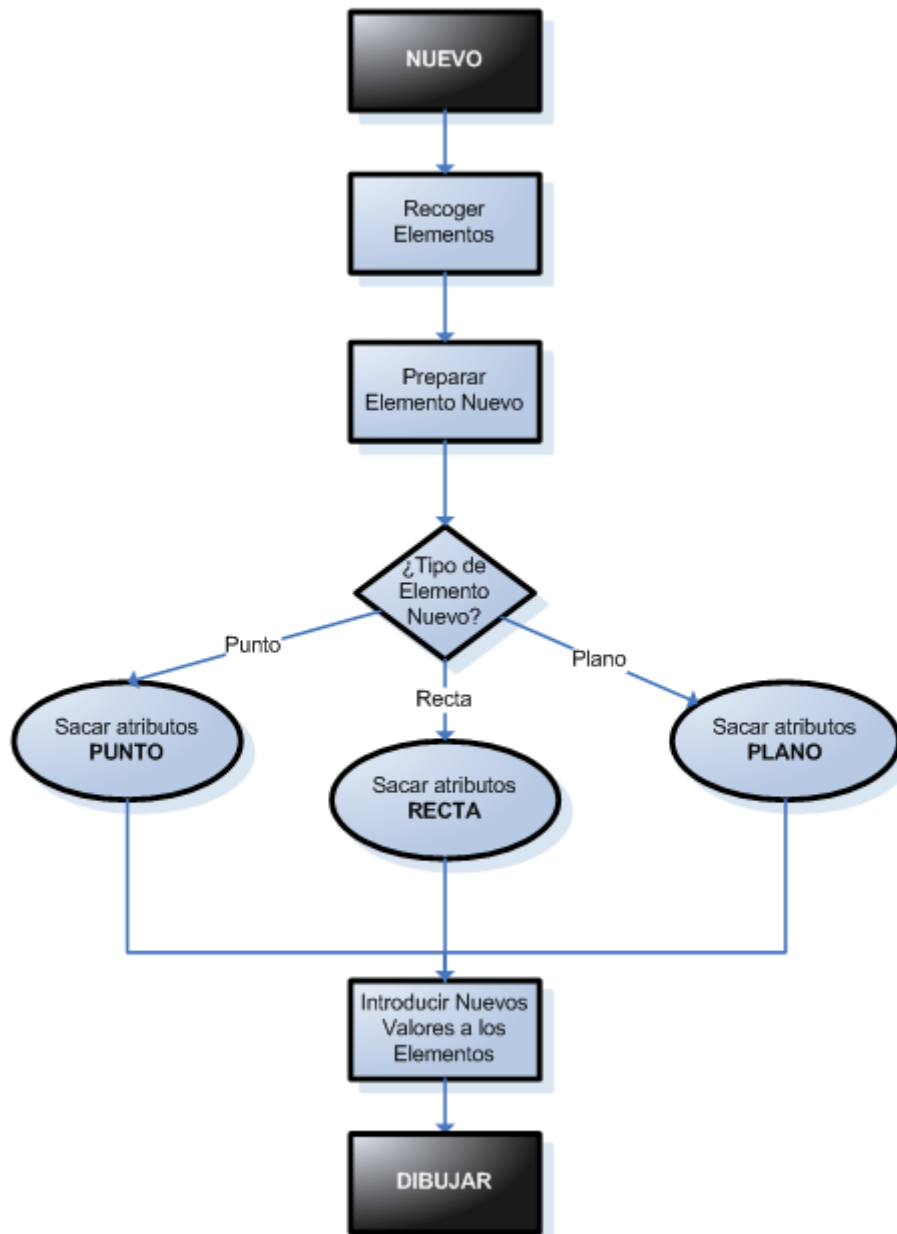


Fig. 7.9 Diagrama de Nuevo

1. Como se ve en la Fig. 7.9 lo primero que hace la función Nuevo es recoger la información de los componentes requeridos, en este caso serán, los elementos ocultos “puntos”, “rectas” y “planos”, donde se guardarán los atributos de cada elemento que se vayan creando, el valor del campo de texto que puede contener la información necesaria para crear el nuevo elemento, los listbox de selección, tanto la selección que tenga cada uno de ellos que indica los componentes que forman el elemento nuevo, como los datos visibles que representan cada elemento que se actualizarán con el elemento nuevo.
2. Previo al calculo de los atributos del elemento, se prepara una cadena de caracteres con la información del elemento.

La cadena debe empezar por “P” si es un punto, por “r” si es una recta y por “a” si es un plano, si la selección en los listbox no puede representar ningún elemento se recoge el valor del campo de texto que, si no está vacío, se toma como el elemento a crear, si su primer carácter no es ninguno de los mencionados, lo interpreta como un punto y le añade una “P” inicial. En el caso de que la selección de los listbox represente elementos, se prepara la cadena de caracteres de la siguiente manera:

- Si hay dos puntos seleccionados → recta: 'r Px Px'
- Si hay tres puntos seleccionados → plano: 'a Px Px Px'
- Si hay un punto y una recta seleccionados → plano: 'a Px rx'
- Si hay dos recta seleccionadas → plano: 'a rx rx'

Siendo la x la posición de la selección menos uno, es decir, Px/rx es el nombre del elemento seleccionado.

3. Cuando se tiene la cadena, en el siguiente paso de la Fig. 7.9, preparar atributos, además de calcular los atributos del elemento y guardarlos a continuación de los de los elementos ya creados, se añade un nuevo elemento en el listbox que corresponda.
4. Por último se guardan los elementos actualizados en la estructura general del programa.

• *Atributos del Punto*

En la Fig. 7.10 se ven los pasos que se realizan para sacar los atributos, los pasos con información útil serán:

1. En primer lugar de la cadena de caracteres introducida deberán sacarse las coordenadas, para ellos se separarán en distintas variables (que corresponden a las coordenadas) los conjuntos caracteres numéricos (números, “-” y “.”) que están juntos.
2. Luego se guardan en la memoria oculta “*puntos*”, a continuación de las coordenadas de los puntos ya introducidos.
3. Por último se introduce el nuevo punto en el elemento de selección, si el punto introducido es el número i, el texto a introducir es: 'Pi(coordenada1, coordenada2, coordenada3)'.

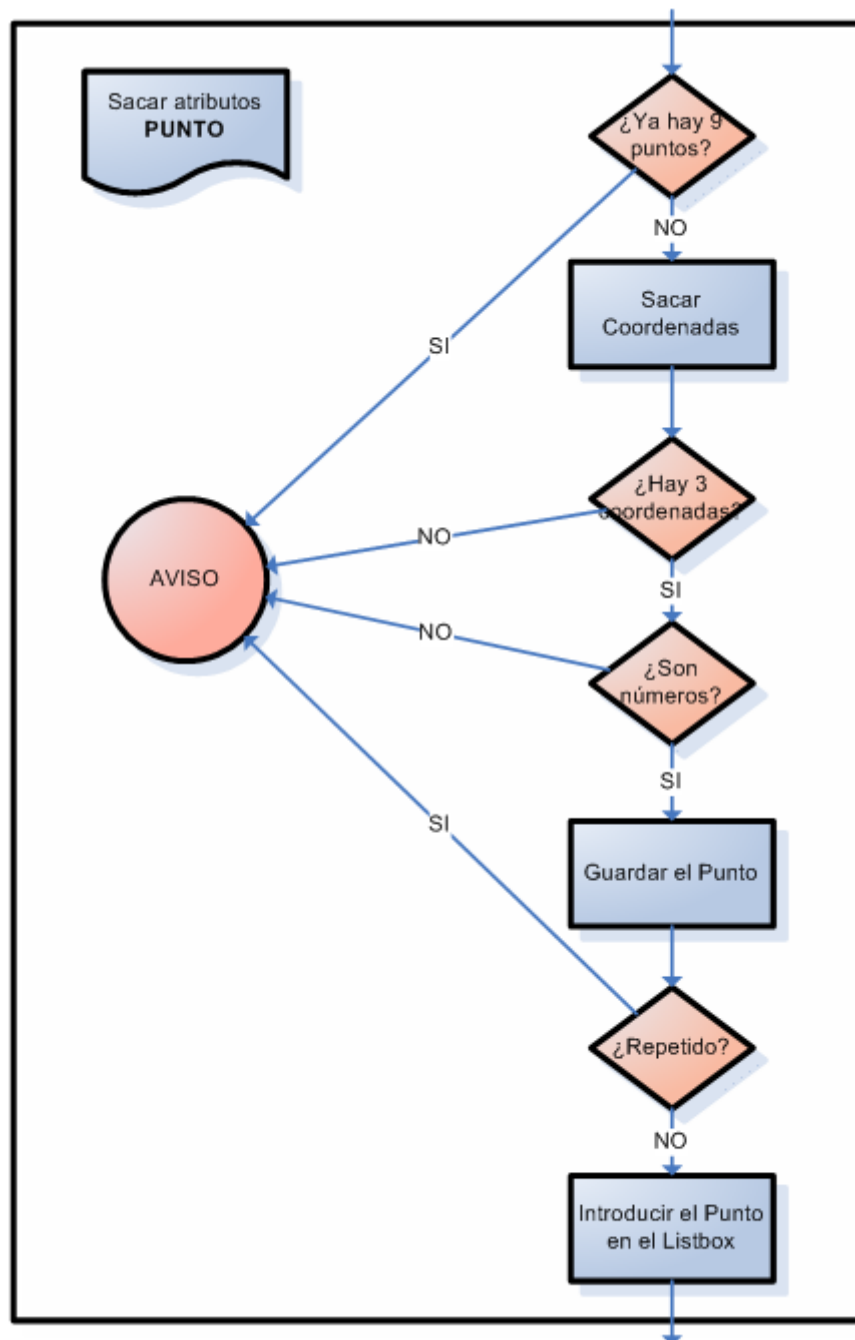


Fig. 7.10 Diagrama de Nuevo: Sacar atributos Puntos

En la Fig. 7.10 también están representados los fallos controlados, que llamarán a la función Aviso para que muestre un mensaje al usuario.

Los avisos comunes son los que aparecen en la Fig. 7.11 que se ha sobrepasado el número máximo de puntos a crear, y el de la Fig. 7.12 que indica que el punto introducido ya está creado.

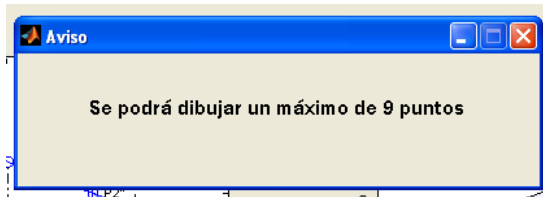


Fig. 7.11 Aviso: Máximo de Puntos Superado

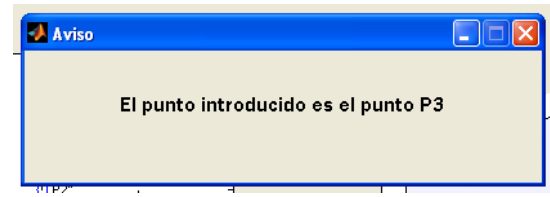


Fig. 7.12 Aviso: Punto Repetido

También se avisará sobre la mala introducción de las escalas, así la Fig. 7.13 muestra el aviso para coordenadas mal introducidas, al separar la cadena en coordenada tiene en cuenta todos los caracteres numéricos, pero no toda las combinaciones forman un número (por ejemplo 12.3.4 no es número). Y en la Fig. 7.14 aparece el aviso de un número incorrecto de coordenadas.

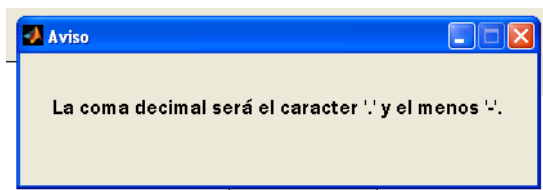


Fig. 7.13 Aviso: Coordenadas no Numéricas

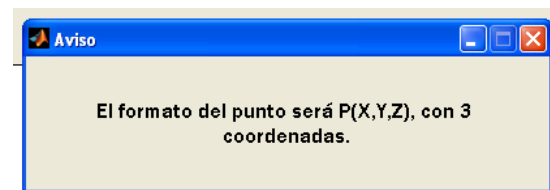


Fig. 7.14 Aviso: Número de Coordenada Incorrecto

- *Atributos de la Recta*

En la Fig. 7.15 se ven los pasos que se realizan para sacar los atributos de la recta, los pasos con información útil serán:

1. En primer lugar, de la cadena de caracteres preparada en el segundo paso de la función Nuevo (representado en la Fig. 7.9), deberán sacarse los puntos que forman la recta, para eso usaremos la función “sacaIndices”, que se explica a continuación.
2. Con los puntos obtenidos se calcula el vector director restando un punto de otro y se guarda en la memoria oculta “rectas” el vector director y las coordenadas de uno de los puntos, a continuación de las rectas ya introducidas.
3. Por último se introduce la nueva recta en el elemento de selección, si la recta introducida es el número i , el texto a introducir es: ‘ri(punto1, punto2)’.

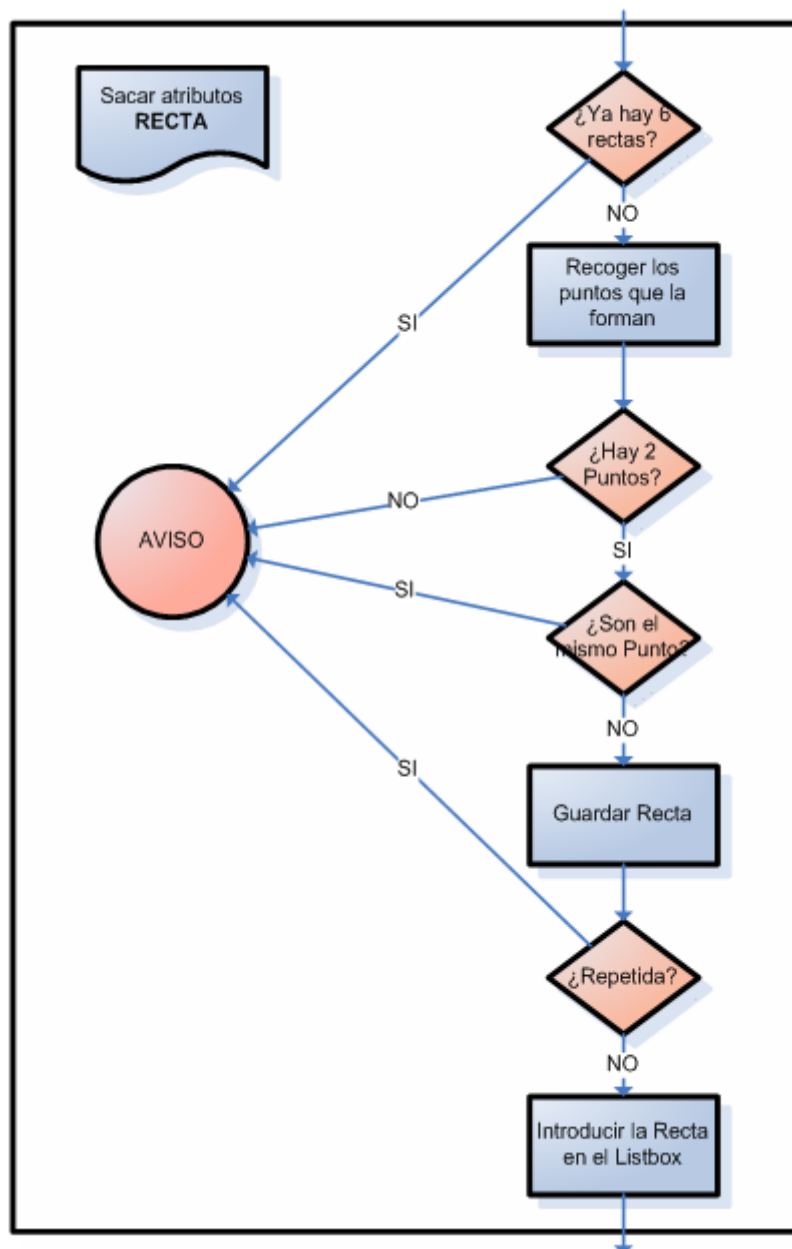


Fig. 7.15 Diagrama de Nuevo: Sacar atributos Rectas

En la Fig. 7.15 también están representados los fallos controlados, que llamarán a la función Aviso para que muestre un mensaje al usuario.

Los avisos comunes son los que aparecen en la Fig. 7.16 que se ha sobrepasado el número máximo de rectas a crear, y en la Fig. 7.17 que indica que la recta introducida ya existe, para comprobar si el vector director y el punto que caracterizan la recta es igual que otra recta con un vector director y un punto distintos se usa la función “*relacionRectas*” que se explicará más adelante.

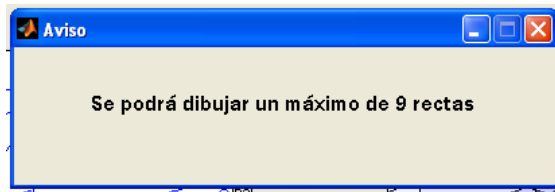


Fig. 7.16 Aviso: Máximo de Rectas Superadas

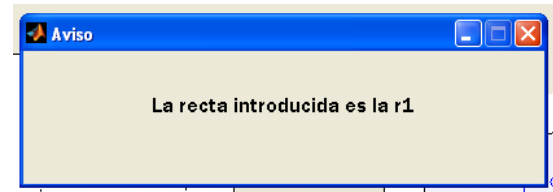


Fig. 7.17 Aviso: Recta Repetido

Además si se utiliza el campo de texto en vez de la selección en el listbox, para introducir los puntos que forma una recta, aparecen más posibilidades de error: se pueden introducir más o menos puntos de los necesarios (aviso reflejado en la Fig. 7.18), o que se introduzcan dos puntos iguales (como en el caso de la Fig. 7.19).

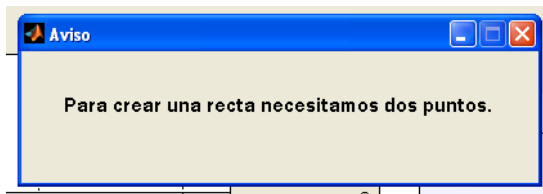


Fig. 7.18 Aviso: Nº de Puntos para Recta Incorrecto

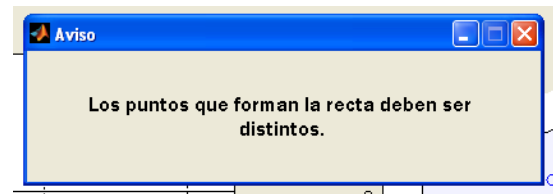


Fig. 7.19 Aviso: Puntos de la Recta Iguales

- *Atributos de la Plano*

En la Fig. 7.20 se ven los pasos que se realizan para sacar los atributos, los pasos con información útil serán:

1. En primer lugar, de la cadena de caracteres preparada en el segundo paso de la función Nuevo (representado en la Fig. 7.9), deberán sacarse los puntos/rectas que forman el plano, para eso usaremos la función “*sacaIndices*”, que se explica a continuación.
2. Con los elementos obtenidos se consiguen tres puntos, en caso de ser un plano creado utilizando rectas, se recogen los dos puntos que se han utilizado para crear esas rectas (usando la función “*sacaIndices*” en lo introducido en el listbox).
3. Con los puntos recogidos, se calculan las constantes de la ecuación paramétrica, que serán las que caracterizan al plano, una vez que las tenemos, se introducen estas cuatro constantes en el elemento oculto “planos”, a continuación de los planos introducidos anteriormente.
4. Por último se introduce el nuevo plano en el elemento de selección. Si el plano introducido es el número i , dependiendo de qué elementos lo formen, la cadena a introducir será: ‘ $ai(punto1,punto2,punto3)$ ’, ‘ $ai(punto1,recta1)$ ’ o ‘ $ai(recta1,recta2)$ ’, esta cadena se habrá montado en el paso dos de recogida de puntos para el cálculo de la ecuación paramétrica como se muestra en la Fig. 7.20.

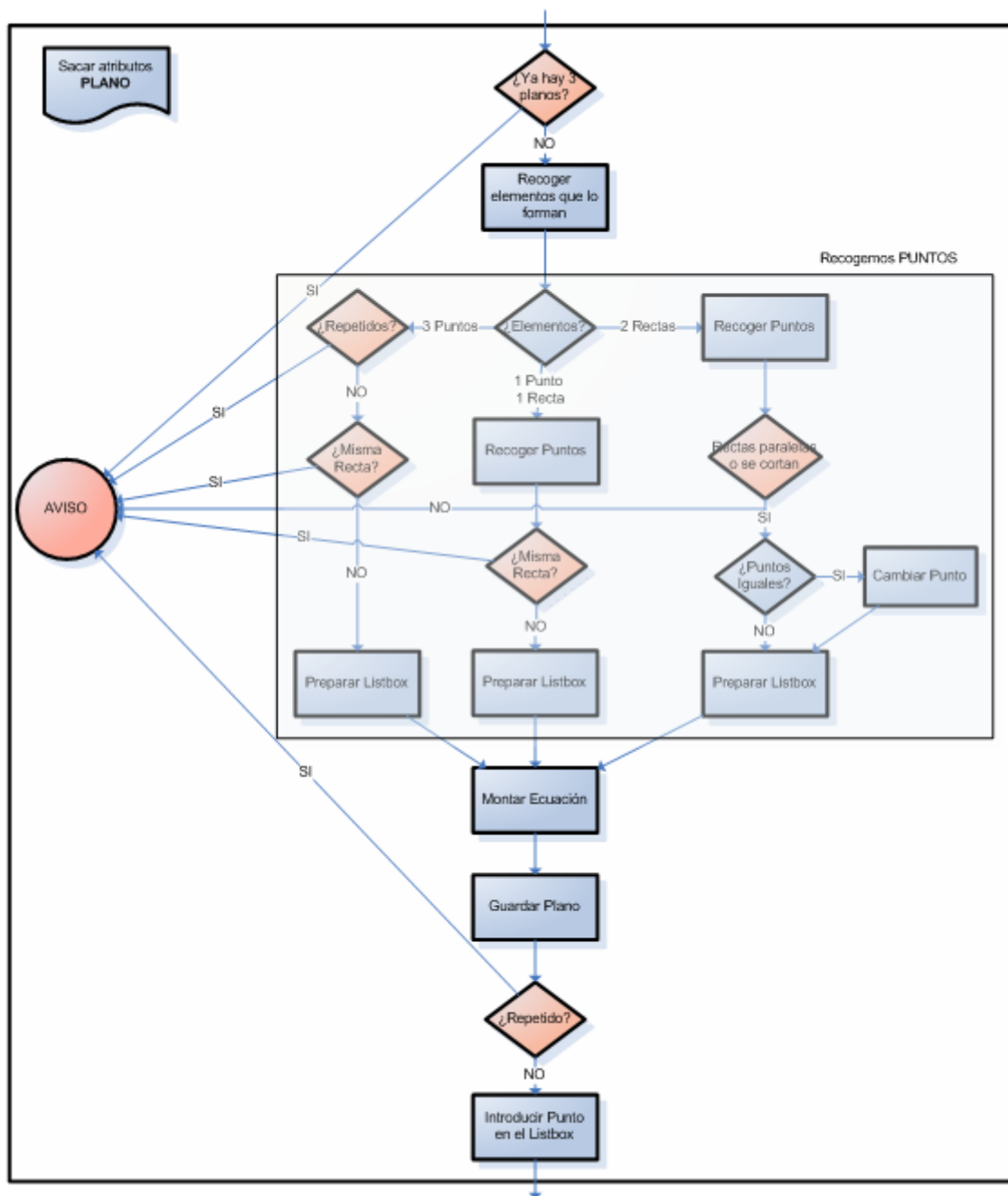


Fig. 7.20 Diagrama de Nuevo: Sacar atributos Planos

En la Fig. 7.20 también están representados los fallos controlados, que llamarán a la función Aviso para que muestre un mensaje al usuario.

Los avisos comunes son los que aparecen en la Fig. 7.21 que indica se ha sobrepasado el número máximo de planos a crear, y en la Fig. 7.22 que indica que el plano introducido ya existe.

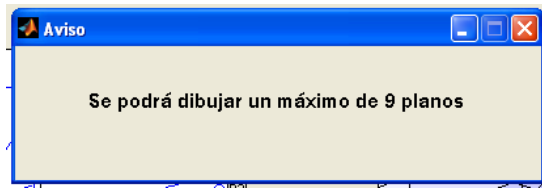


Fig. 7.21 Aviso: Máximo de Planos Superados

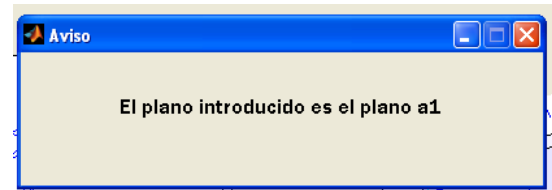


Fig. 7.22 Aviso: Plano Repetido

En el caso de ser un plano creado a partir de tres puntos puede pasar que los puntos estén repetidos, apareciendo el aviso de la Fig. 7.23 (posible sólo si se introducen los planos a partir del campo de texto), o que sean tres puntos alineados y formen una recta, con lo que no se podrá crear un plano único, apareciendo el aviso de la Fig. 7.24.

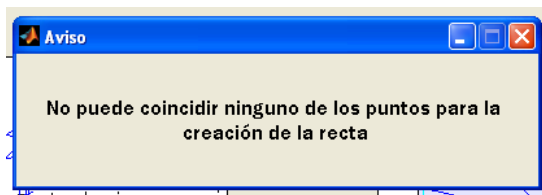


Fig. 7.23 Aviso: Puntos para Crear Planos Repetidos

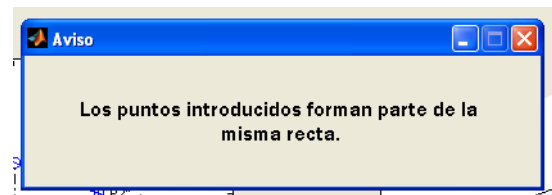


Fig. 7.24 Aviso: Puntos Forman la Misma Recta

Si el plano se crease a partir de una recta y un punto, se tendrá en cuenta que el punto no forme parte de la recta. Esto se controlará con el aviso de la Fig. 7.25.

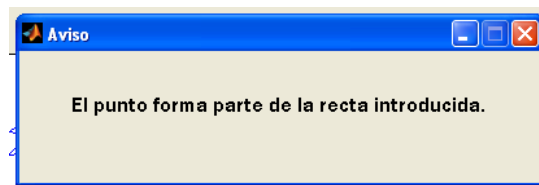


Fig. 7.25 Aviso: Punto Dentro de la Recta

Dos rectas pueden crear un plano si son paralelas o se cortan, se comprobará con la función “*relacionRectas*” que se explicará más adelante. En caso de no formar un plano aparece el aviso de la Fig. 7.26.

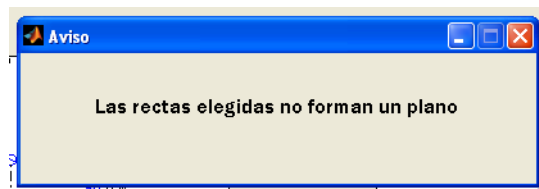


Fig. 7.26 Aviso: Las Rectas no forman un Plano

7.3.3. Borrado de Elementos: Función BORRAR

La función Borrar, se encarga de borrar de las memorias los elementos seleccionados, como parámetro de entrada sólo tiene la estructura o puntero que se ha formado al iniciar el programa de los componentes que se modifican. Al final la función llama a Dibujar si hay elementos a dibujar o a Limpiar si no los hay.

Esta función tiene tres objetivos:

- Eliminar los elementos de la memoria visible.
- Eliminar los elementos de la memoria oculta.
- Renombrar los elementos del mismo tipo de los elementos a borrar con un índice mayor y cambiar el nombre en las referencias a ellos en los elementos de otro tipo.

En esta función se recogen los elementos a borrar seleccionados por el usuario, por lo que puede ser inválida produciendo los llamados “errores de inserción” que son controlados por el programa y se informan en forma de avisos al usuario.

En la Fig. 7.27 se ve la secuencia general que sigue la función Borrar:

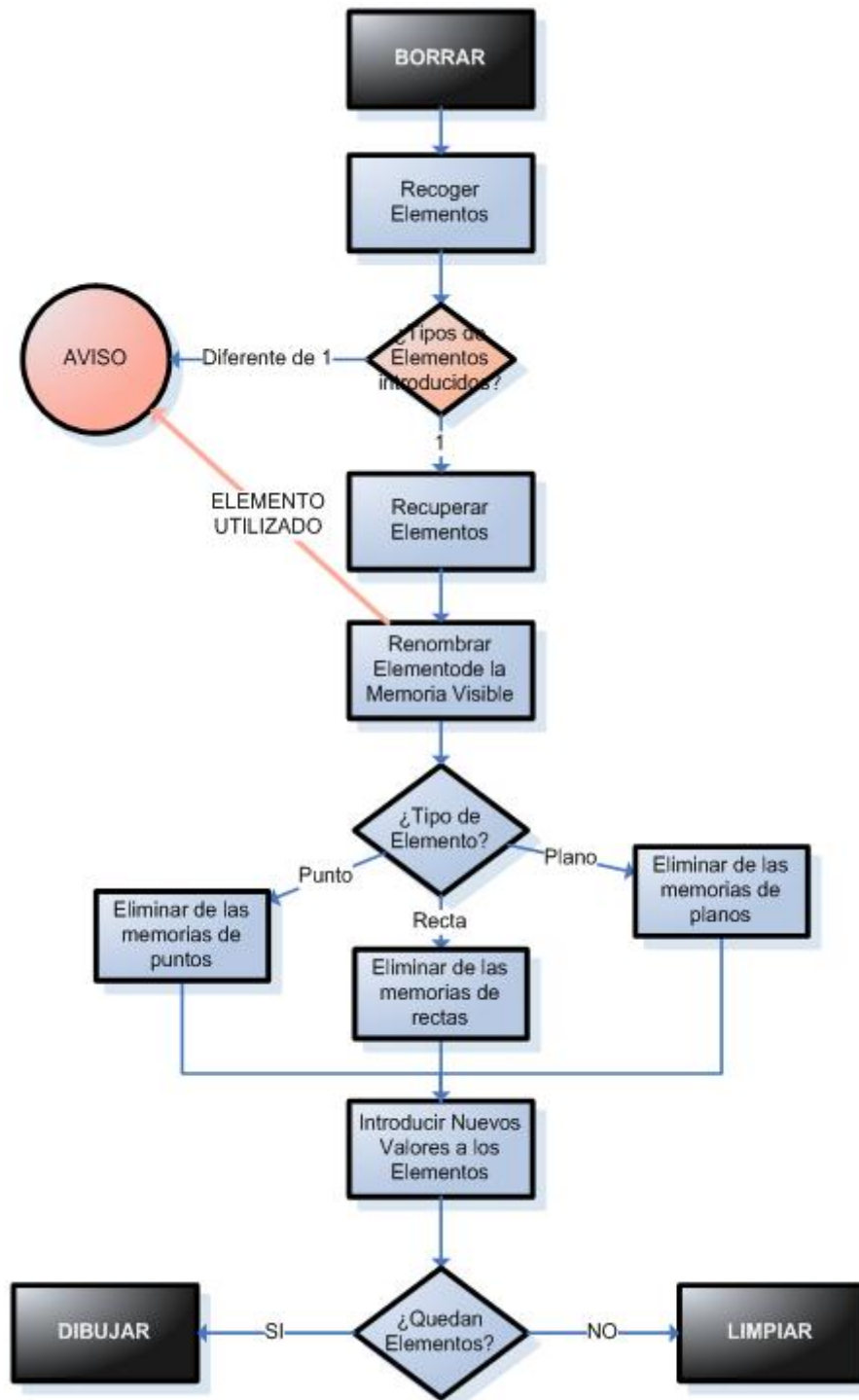


Fig. 7.27 Diagrama de Borrar

Para conseguir el borrado del elemento se realizan los pasos reflejados en la Fig. 7.27

1. Se recuperan los elementos: En este caso se necesitan las memorias ocultas, los valores de los listbox de selección, que informa del elemento a borrar y de la memoria visible del listbox que se tendrán que cambiar al renombrar los elementos.
2. Renombrar la memoria visible: A los elementos se les da nombre según se van creando. Este nombre estará formado por el carácter que refleja el tipo de elemento y

el número que le corresponda por orden de creación, es decir, el siguiente al del elemento del mismo tipo creado antes que éste, por lo que al borrar, queda un número en el aire. Para evitarlo se renombrarán los elementos del mismo tipo siguientes a él, y por extensión, la referencia que se haga a ellos en los elementos de otro tipo. Esto se hace llamando a la función *“borrarAuxiliar”*, que se explica a continuación, esta función es llamada tres veces, una por cada tipo de elemento, para cambiar los nombres de su memoria visible.

3. Se elimina el elemento seleccionado de la memoria oculta y visible correspondiente. Para la posición de selección i (elemento $i-1$) en el caso del tipo de elemento:

Punto: Se elimina de la memoria oculta *“puntos”* desde la posición $(3i - 5)$ hasta la $(3i - 3)$, y de la memoria visible *“listpuntos”* el elemento seleccionado.

Recta: Se elimina de la memoria oculta *“rectas”* desde la posición $(6i - 11)$ hasta la $(6i - 6)$, y de la memoria visible *“listrectas”* el elemento seleccionado.

Plano: Se elimina de la memoria oculta *“planos”* desde la posición $(4i - 7)$ hasta la $(4i - 4)$, y de la memoria visible *“listplanos”* el elemento seleccionado.

4. Antes de llamar a otra función se introducirán los elementos modificados en la estructura, donde se guardarán para futuras ejecuciones.

Una vez terminados todos los pasos necesarios para borrar los elementos se llama a la función *dibujar*, para que dibuje los elementos que permanezcan sin borrar, en caso de no haber, se llama a la función *Limpiar*, que reinicia el espacio de trabajo.

En la Fig. 7.27 también están representadas las llamadas a la función *Aviso* para que muestre un mensaje al usuario con los fallos controlados. Un error posible es la introducción de más de un tipo de elemento a borrar, controlado por el aviso de la Fig. 7.28.

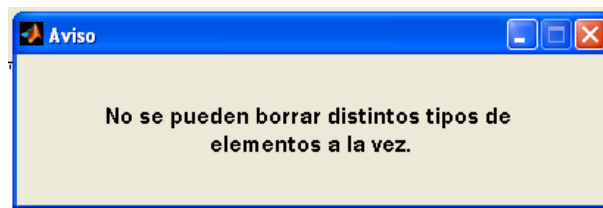


Fig. 7.28 Aviso: Varios Elementos Seleccionados

Además la función *“borrarAuxiliar”* encargada de renombrar elementos, también comprueba que el elemento a borrar no esté siendo utilizado para montar otro elemento, si fuese así devuelve un mensaje de error, que se muestra como el aviso, de la Fig. 7.29.

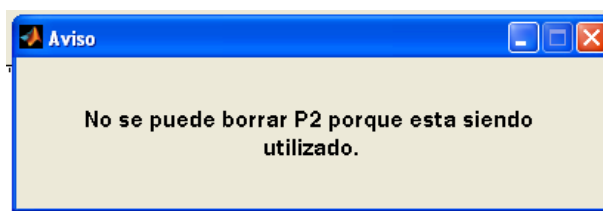


Fig. 7.29 Aviso: Elemento utilizado

Función *borrarAuxiliar*

La función “*borrarAuxiliar*” es un complemento de la función principal *Borrar*, que actúa sobre los elementos de las memorias visibles. Aparece por la necesidad de renombrar los elementos del mismo tipo que el elemento a borrar. Por ejemplo si hay cinco puntos creados, estos serán P1, P2, P3, P4 y P5, si se borra el punto P2, los puntos P3, P4 y P5 deben pasar a ser P2, P3 y P4, si además, estos puntos se han utilizado para crear otro elemento, por ejemplo un plano que inicialmente se muestra al usuario como ‘a1(P3,P4,P5)’, pasará a mostrarse como ‘a1(P2,P3,P4)’.

Esta función tiene dos objetivos:

- Valida que el elemento a borrar no haya sido utilizado para la creación de otro elemento.
- Renombra los elementos del mismo tipo, cambiando la su parte numérica o índice, es decir, resta uno del índice, de los elementos del mismo tipo, cuyo índice es mayor que el del elemento introducido.

La entrada a esta función depende de datos introducidos por el usuario, por lo que puede encontrar inconsistencias controladas con un mensaje de aviso como salida.

La inicialización de esta función es: [mem, txt] = *borrarAuxiliar*(mem, carini, indice), donde se puede observar que tiene dos variables de salida y tres variables de entrada:

VARIABLES DE ENTRADA

mem: Matriz de caracteres que contiene la memoria visible a cambiar.

carini: Carácter (‘P’, ‘r’ o ‘a’) del tipo de elementos a renombrar.

indice: Variable numérica que corresponde al Índice a partir del que se deben renombrar los elementos, es decir, valor numérico que forma parte del nombre del elemento que se ha introducido para borrar el función principal *Borrar*.

VARIABLES DE SALIDA

txt: Cadena de caracteres que en caso de error por datos de entrada contendrá el mensaje de error.

mem: Memoria de salida. Este campo es a la vez variable de entrada y de salida, ya que es la variable a actualizar

El nombre completo del elemento base, es decir, del elemento que en este momento se borrará, será la unión de las variables de entrada carini e indice. Todos los índices de los elementos del tipo introducido mayores a este valor son los que se cambiarán.

Se llama a esta función solamente desde la función principal *Borrar*, una vez por cada una de las memorias visibles, para que cambie los nombres de los elementos que deban cambiarse, pasándole el elemento que se va a borrar, y la memoria a renombrar.

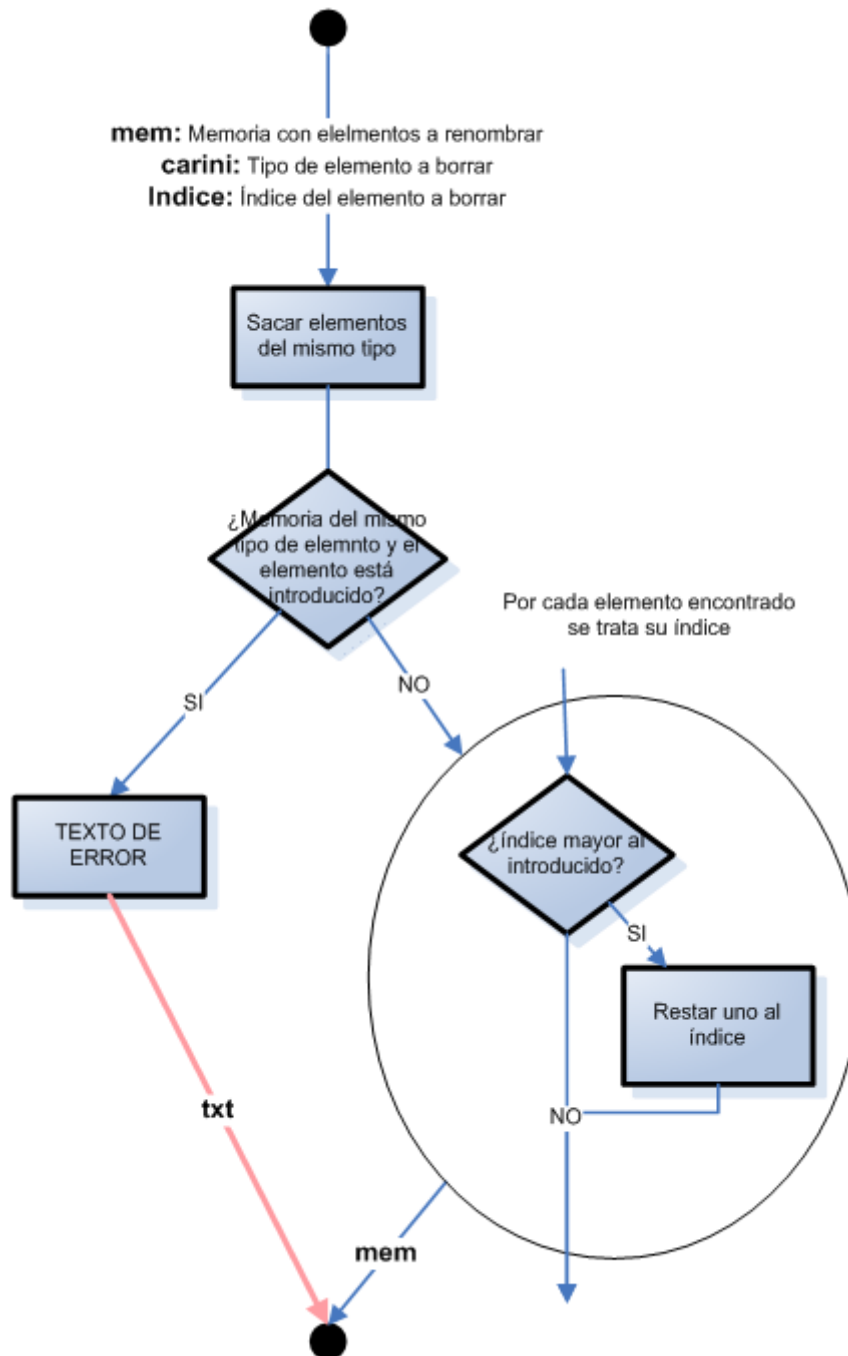


Fig. 7.30 Diagrama Función borrarAuxiliar

En la Fig. 7.30 se ve la los pasos necesarios para conseguir los objetivos:

1. De la memoria introducida (mem) se sacan todos los elementos del tipo que indica la variable carini con la función "sacaIndices",
2. Si la memoria introducida no es la memoria del tipo de elemento, y está el elemento que forman carini e índice, termina la función enviando un mensaje de error que deberá mostrarse en un aviso (en la Fig. 7.29)
3. Por cada elemento conseguidos en el primer paso, se comprueba si el índice es mayor que el de la variable de entrada, y si lo es se resta uno.

7.3.4. Dibujar Elementos: Función DIBUJAR

La función Dibujar, se encarga de hacer las representaciones, tanto en dos como tres dimensiones, los parámetros de entrada son la estructura o puntero que se ha formado al iniciar el programa de los componentes que se modifican, y una variable que indica desde donde se le llama, para distinguir entre dibujar todos los elementos, elementos seleccionados o utilizar la memoria de la ejecución anterior para repetirlo en otra figura.

En la Fig. 7.31 se ve la secuencia general que sigue la función Dibujar:

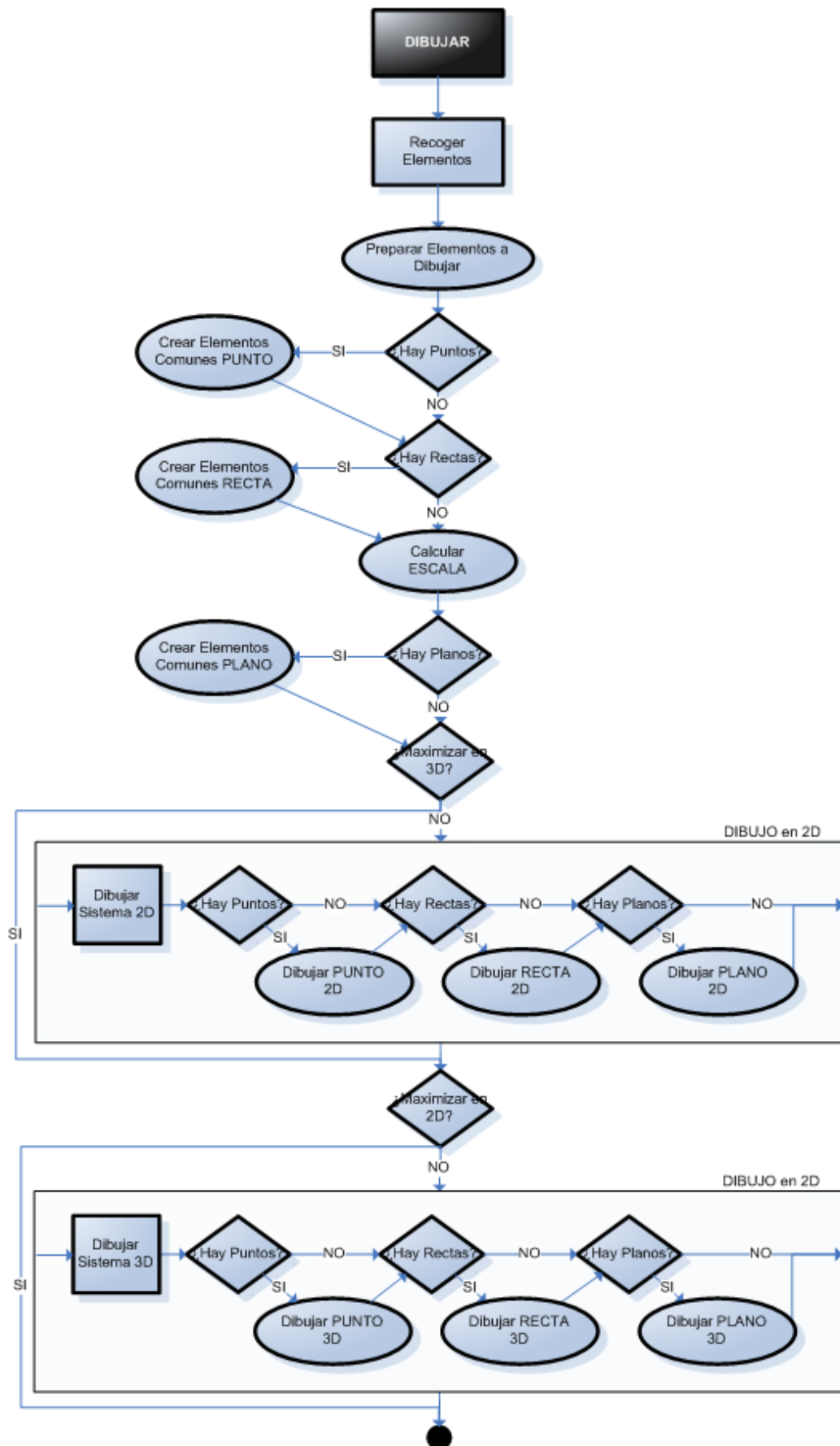


Fig. 7.31 Diagrama de Dibujar

Como se ve en la Fig. 7.31 antes de realizar el dibujo, se necesitan unos pasos previos:

- Como en todas las funciones principales, primero se recogen los componentes necesarios de la estructura.
- Después se preparan los elementos a dibujar, esto depende del lugar desde donde se llame la función y de lo que se pretenda dibujar.
- Con lo obtenido se calculan las escalas y se crean los elementos comunes necesarios para dibujar tanto en tres como en dos dimensiones, el caso del punto y de la recta se harán antes de calcular las escalas ya que los necesitan, y el caso de los planos se hace después porque utilizan las escalas para el cálculo.

Los componentes requeridos, en este caso, serán, el checkbox *“auxil”* que indicará si se dibujan o no las líneas auxiliares, los elementos ocultos *“puntos”*, *“rectas”* y *“planos”*, que contienen las características necesarias para crear cada elemento, el elemento oculto *“antele”* que en el caso de haber llamado a la función desde algunas de las lupas tiene todo lo necesario para repetir el dibujo de la ejecución anterior y en caso de no venir de una lupa guardar los datos que permitirían repetir el dibujo en una futura ejecución.

Preparar los Elementos a Dibujar

Para preparar los elementos a dibujar como muestra en la Fig. 7.32 primero se comprueba si viene de maximizar para simular la ejecución anterior y una vez simulada o no, se preparan los elementos, ya sean todos o sólo los dibujados.

En este paso cobra vital importancia el componente oculto *“antele”*, ya que para repetir dibujos tiene todos los datos necesarios para la reproducción, y además en guarda los textos a mostrar.

Se crea en las al llegar de la función Nuevo o Borrar, o de pulsar el botón *“Dibujar”*, por cada elemento que dibuja se guarda una fila con el posición en el listbox del elemento (nombre +1) y una *‘t’*, por ejemplo si se dibuja el punto P1 uno de los registros tiene el valor *‘P2t’*, la *‘t’* significa que de ese elemento se va a crear texto.

Al venir de textos cambia las *‘t’* por espacios *‘ ’*, o si los espacios *‘ ’* por *‘t’*, al venir de un listbox, del elemento seleccionado se cambia este carácter si tiene *‘t’* por *‘ ’* y viceversa.

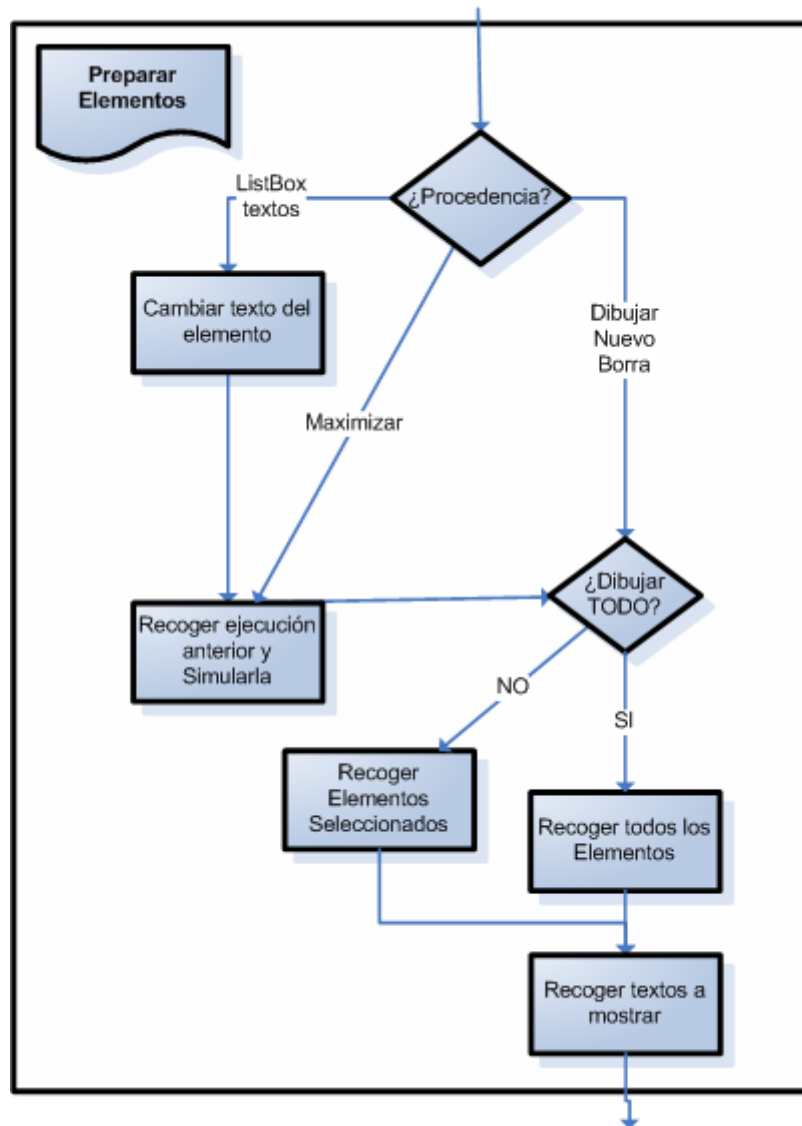


Fig. 7.32 Diagrama de Dibujar: Preparar Elementos

Si se van a dibujar todos los elementos:

Se guardarán en los array de puntos, rectas y planos a dibujar todos los que estén en los elementos ocultos de “*puntos*”, “*rectas*” y “*planos*”.

En el componente oculto “*antele*” se guarda la variable de entrada que indica desde donde se llama y el valor del check “*auxil*” y todos los elementos, que se indicarán con una “*P*”, “*r*” o “*a*”, dependiendo del tipo de elemento y la posición del listbox, es decir, no será en nombre del punto ya que la posición es el índice del nombre más uno, seguido de una ‘*t*’.

Si se quiere dibujar el conjunto de elementos seleccionados:

Se guardarán en los array de puntos, rectas y planos a dibujar los elementos seleccionados en los listbox. Si el primer elemento de uno de los listbox está seleccionado, se tomará como no selección de ningún elemento de ese listbox. Para

recoger las características del elemento de su memoria visible se tendrá en cuenta la correspondencia que hay entre las posiciones, explicadas al inicio de este apartado:

- Para la posición i de “*listpuntos*”, es decir el punto $i-1$, las posiciones en la memoria oculta serán desde $(3i - 5)$ hasta $(3i - 3)$.
- Para la posición i de “*listrectas*”, es decir la recta $i-1$, las posiciones en la memoria oculta serán desde $(6i - 11)$ hasta $((6i - 6)$.
- Para la posición i de “*listplanos*”, es decir el plano $i-1$, las posiciones en la memoria oculta serán desde $(4i - 7)$ hasta $(4i - 4)$.

En el componente oculto “*antele*” se guarda la variable de entrada que indica desde donde se llama la función, el valor del check “*auxil*” y los elementos a dibujar, que se indicarán con una “P”, “r” o “a”, dependiendo del tipo de elemento y la posición del listbox, es decir, no será en nombre del punto ya que la posición es el índice del nombre más uno, seguido de ‘t’.

Si se quiere maximizar uno de los dibujos:

Con el elemento oculto “*antele*” se simularán las mismas variables, valores y propiedades que el caso de la ejecución anterior, indicando en otras variables distintas su procedencia, de tal forma que si se quiere dibujar todos los elemento sólo los seleccionados en el caso anterior se hará exactamente igual que lo explicado en los ortos dos puntos, con la única diferencia de que el componente “*antele*” no se modificará, para que siga guardando las mismas características.

Si se quiere cambiar los textos:

Para cambiar los textos se accede a la ‘t’ del elemento guardada en “*antele*” y se cambia por espacio ‘ ’, si se quiere eliminar, en caso contrario, no habrá ‘t’ sino un espacio ‘ ’ que se cambia por ‘t’. Y luego se simulará el dibujo anterior de igual manera que cuando se quiere maximizar uno de los dibujos.

Una vez seleccionado los elementos y creado o modificado el elemento oculto si es necesario, venga de donde se venga a la función Dibujar, se guardan los elementos que deben tener textos identificativos, a través de la ‘t’ o ‘ ’ creados por cada elemento a dibujar. Para la llegada a la función por el botón “Dibujar” o las funciones Nuevo y Borrar se dibujan todos los textos.

Crear Elementos Comunes

Hay elementos necesarios para el dibujo, que son comunes para dos y tres dimensiones, y además tendrán una estrecha relación con las escalas.

Punto

Para el punto se crean los textos identificativos de los puntos y sus proyecciones, en este caso no hay ninguna influencia sobre las escalas.

Recta

Para la recta se calculan sus trazas. Cuando el usuario seleccione alguna recta para dibujar se mostrarán todos los elementos de esta (lo que incluye a las trazas), es decir, las trazas deberán estar dentro de la zona visible, por eso este apartado debe estar antes de que el cálculo de las escalas, ya que para estos casos las escalas dependen de ellas.

Plano

En lo que se refiere al plano, sólo hay cálculos para el caso de que este pase por la Línea de Tierra, ya que las trazas no son información suficiente, haciéndose necesario un punto perteneciente al plano, este punto con su texto se introduce en los puntos a representar, teniendo en cuenta que su texto se muestran si se muestran los del plano. Para la elección de este punto se utilizan las escalas, por lo cual, se hará después del cálculo de las escalas.

Cálculo de Escalas

Para el cálculo de escalas lo primero que se debe hacer es decidir qué valores de X, Y y Z se van a ver. En todo caso las coordenadas de los puntos deben estar visibles, así como los puntos de corte de los planos (sin características especiales) con la Línea de Tierra. En caso de dibujarse por selección, también tendrán que ser visibles las trazas de las rectas que los definen.

Se calculan las escalas de cada una de las coordenadas de forma independiente, creándose arrays diferentes para todas las X, Y, Z que deben entrar dentro del dibujo, y se actuará con cada uno de ellos por separado.

Para las X's, se harán más validaciones, ya que es la coordenada que más afecta a la visión global del dibujo en dos dimensiones. Los límites en este caso dependen del número de X's no repetidas, por lo que primero se utilizará la función "*noRepetido*" cuya salida es el array de entrada sin ningún elemento repetido ordenado de mayor a menor. Una vez que se tienen todas las posibles X's ordenadas se calcula el incremento a realizar, si hay menos de cuatro puntos el incremento es la media del máximo y el mínimo, en caso contrario será constante, uno. Por último se comprueba que, con el incremento de escala elegido, la separación entre x sea al menos de 1/10 de la escala sino pone el incremento un incremento mínimo.

Las Y's y las Z's tendrán el mismo tratamiento, primero se calcula un incremento proporcional al valor absoluto máximo de las Y's, o Z's, y luego se le aplicará el incremento según su valor. Es un método mucho menos rígido que para las X's, ya que se le ha dado más importancia a éstas.

- **Re-calcular escalas**

Un plano no tiene límites, es infinito, pero para dibujarlo se necesitan límites. La representación del plano en tres dimensiones se hace dibujando un paralelogramo que se calcula a partir de las escalas, con la función "*limitesPlano*" que se explicará a continuación, en ella se calculan los puntos límite del dibujo de las trazas del plano a partir de las escalas, en la Fig. 7.33 son los puntos tr1, tr2, tr3, tr4, a partir de los cuales haciendo paralelas se dibuja el paralelogramo que representa el plano.

El problema que se plantea es que al poner límites a un espacio ilimitado puede ocurrir que un punto que forme parte de ese plano se dibuje fuera de su representación, dando lugar a una dificultad, incluso incoherencia, del dibujo, un ejemplo (no real) es el de la Fig. 7.33, en la cual el punto P pertenece al plano α , pero en su representación parece estar fuera.

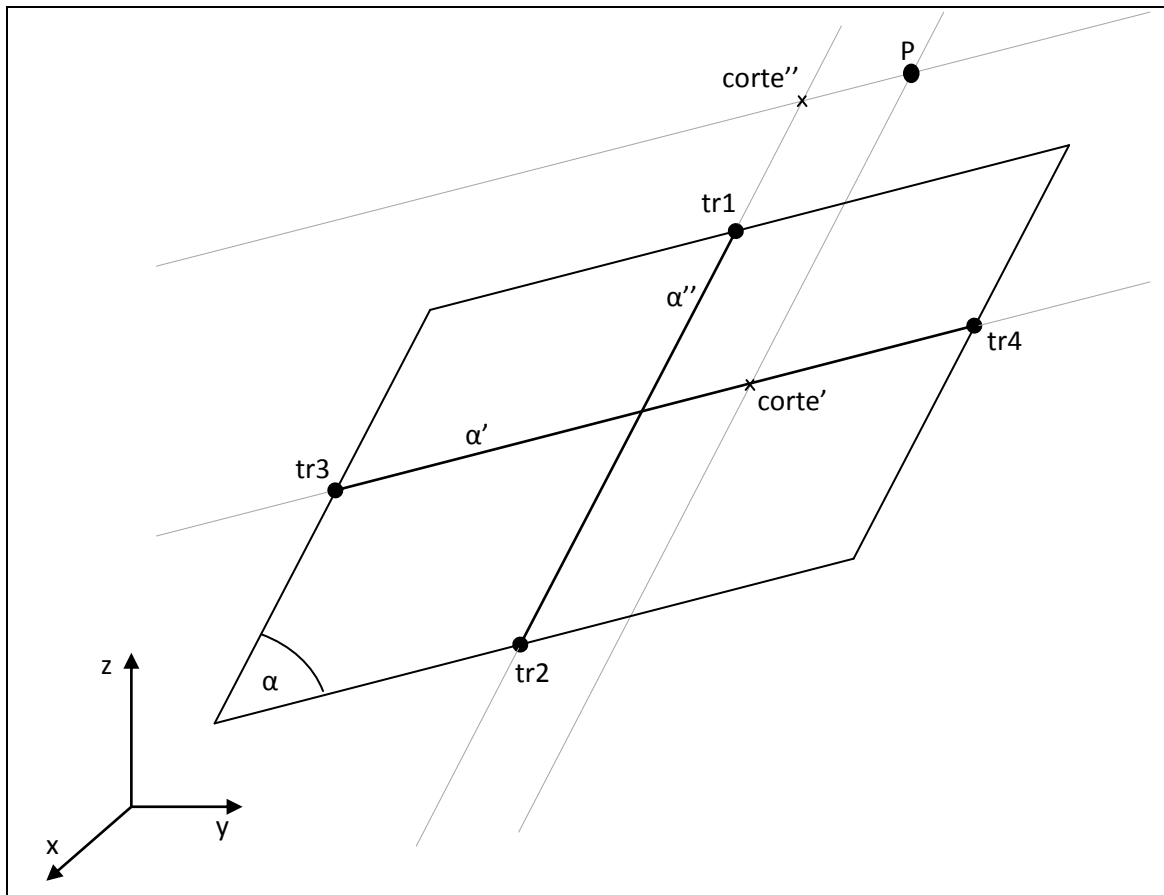


Fig. 7.33 Reajuste de Escalas

Para resolver este problema se cambian las escalas, de forma que se amplía el dibujo del paralelogramo para que el punto se dibuje dentro de la representación.

En primer lugar, será necesario comprobar que los puntos del plano que se dibujan, se encuentran dentro de su representación. La Fig. 7.33 representa la forma de comprobar si un punto del plano está dentro de su representación, para ello, se necesitan rectas paralelas a las trazas que pasen por el punto, en este caso P, la recta paralela a la traza α'' corta a la traza α' , en el punto corte' y, la recta paralela a la traza α' corta a la traza α'' , en el punto corte''. Los puntos de corte se calculan con la función *"relacionRectas"* que se explica más adelante, que como su propio nombre indica devuelve la relación entre dos rectas. Si el punto corte' no está entre tr3 y tr4 o el punto corte'' no está entre tr1 y tr2 significa que hay que cambiar las escalas porque el punto está fuera del plano. Ahora bien, se necesita saber cuánto y en qué coordenada se debe aumentar la escala.

En el caso de que el punto corte'' no esté dentro del segmento dibujado de la traza α'' , se necesita alargar la traza α'' . Puede no pertenecer al segmento por la coordenada X o por la coordenada Z, ya que en las trazas verticales la coordenada Y siempre es 0, si es por la coordenada X se aumentará la escala de X hacia la misma dirección y la misma cantidad que la diferencia entre corte'' y el extremo más cercano del segmento dibujado de la traza α'' y, si es por la coordenada Z se aumentará la escala de Z hacia la misma dirección y la misma cantidad que la diferencia entre la Z del corte'' y la Z del el extremo más cercano del segmento dibujado de la traza α'' .

En el caso de que sea el punto corte' el que no esté dentro del segmento dibujado de la traza α' , se necesita alargar la traza α' . Puede no pertenecer al segmento por la coordenada X o por la coordenada Y, ya que en las trazas horizontales la coordenada Z siempre es 0, si es por la coordenada X se aumentará la escala de X hacia la misma dirección y la misma cantidad, que la diferencia entre la X del corte' y la X del el extremo más cercano del segmento dibujado de la traza α' y, si es por la coordenada Y se aumentará la escala de Y hacia la misma dirección y la misma cantidad que la diferencia entre la Y del corte' y la Y del el extremo más cercano del segmento dibujado de la traza α' .

Dibujar Elementos del Sistema

Los dibujos de elementos se pueden hacer en la zona del espacio de trabajo habilitada o en una figura nueva más grande en el caso que se presionen los botones de maximizar, tanto en dos o tres dimensiones, por lo que previo al dibujo de los elementos del sistema, se comprueba si se ha presionado alguno de estos botones, y si es así, se debe abrir una figura más grande donde se hará el dibujo. Los elementos del sistema a dibujar son los mismos que se dibujan en la función Limpiar, pero adaptándolos a las escalas calculadas.

- *En dos dimensiones*

La representación de los elementos del sistema en dos dimensiones se reduce a la Línea de Tierra: una línea horizontal que va de extremo a extremo del dibujo y dos líneas cortitas en cada uno de los extremos

- *En tres dimensiones*

El dibujo de los elementos del sistema en tres dimensiones consiste en los planos de proyección y la línea donde cortan, que corresponde con la Línea de Tierra.

Dibujar Puntos

- *Puntos en dos dimensiones*

Líneas Fijas:

- **Proyecciones del Punto**

Los puntos en dos dimensiones se representan con sus dos proyecciones, que se dibujan a partir de sus coordenadas, una de las proyecciones con las coordenadas X, -Y (proyección horizontal) y la otra con X, Z (proyección vertical).

En la Fig. 7.34 se ven ejemplos de diferentes puntos en dos dimensiones representados con la aplicación, el punto P1 está en el primer cuadrante, P2 en el tercero, P3 en el cuarto, P4 en el segundo, P5 en el Plano de proyección Vertical, P6 en el Plano Horizontal y P7 en la Línea de Tierra.

- **Línea perpendicular a la Línea de Tierra**

Sólo se dibuja en caso de no coincidir con proyecciones de una recta. Para dibujar la línea perpendicular a la Línea de Tierra se necesitan las posiciones relativas entre cada una de las proyecciones con la Línea de Tierra. En el caso de estar una por encima y otra por debajo la línea une las dos proyecciones, en el caso de estar las dos por encima, o las dos por debajo, la línea va desde la Línea de Tierra hasta el punto más lejano de ella, en la Fig. 7.34 se ven puntos con diferentes posiciones relativas.

- **Textos de las Proyecciones (si se selecciona)**

Los textos de las proyecciones de los puntos ya preparados al crear elementos comunes del punto, se colocarán con las mismas coordenadas que sus proyecciones, menos en el caso que las proyecciones verticales y horizontales coincidan, que se colocarán uno al lado del otro como es el caso de los puntos 3 y 7 de la Fig. 7.34.

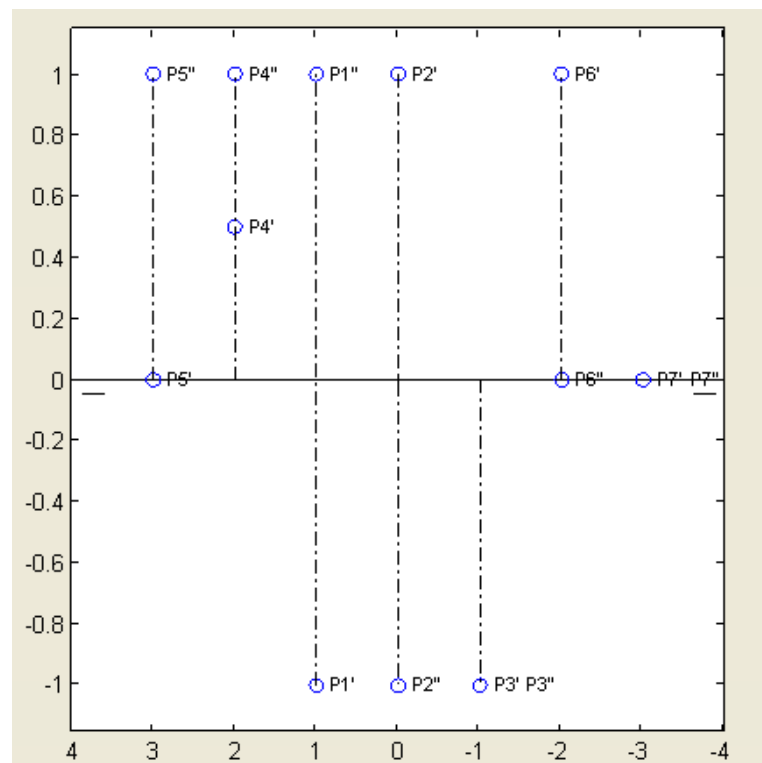


Fig. 7.34 Ejemplo: Puntos 2D

En este caso no habrá diferencias entre el dibujo con líneas auxiliares y sin ellas.

• *Puntos en tres dimensiones*

Líneas Fijas:

- **El Punto**

Para dibujar un punto en tres dimensiones sólo hacen falta las tres coordenadas.

- **Texto del Punto** (si se selecciona)

Los textos de los puntos ya preparados al crear elementos comunes del punto, se colocarán en las coordenadas del punto. Los textos se colocan al lado de la proyección correspondiente, si estas se superponen sus textos se colarán uno al lado del otro

Líneas auxiliares. Se dibujan cuando se viene desde la función genérica Dibujar, y se selecciona la representación con líneas auxiliares. Además de lo mencionado sin líneas auxiliares, se dibujan:

- **Proyecciones del Punto**

Por cada punto se dibujan sus dos proyecciones. La vertical usará las coordenadas X e Y del punto con $Z=0$, y la horizontal usa las coordenadas X y Z con $Y=0$.

- **Líneas de unión del Punto con las Proyecciones**

Se une el punto con las proyecciones mediante dos líneas: una que va del punto al punto con las mismas coordenadas pero con $Y=0$ (proyección sobre el plano horizontal), y la otra del punto a otro con las mismas coordenadas pero con $Z=0$ (proyección sobre el plano vertical).

- **Textos de las Proyecciones** (si se selecciona)

Los textos ya están preparados y se sitúan en los mismos puntos donde se dibujaron las proyecciones, siempre y cuando la proyección no coincida con el punto, es decir que no esté en el plano de proyección, así, en punto P2 de la Fig. 7.35 no se nombra la proyección vertical y en el punto P3, que está en la Línea de Tierra, ninguna de ellas.

En la Fig. 7.35 se ve la representación de un punto en tres dimensiones con líneas auxiliares:

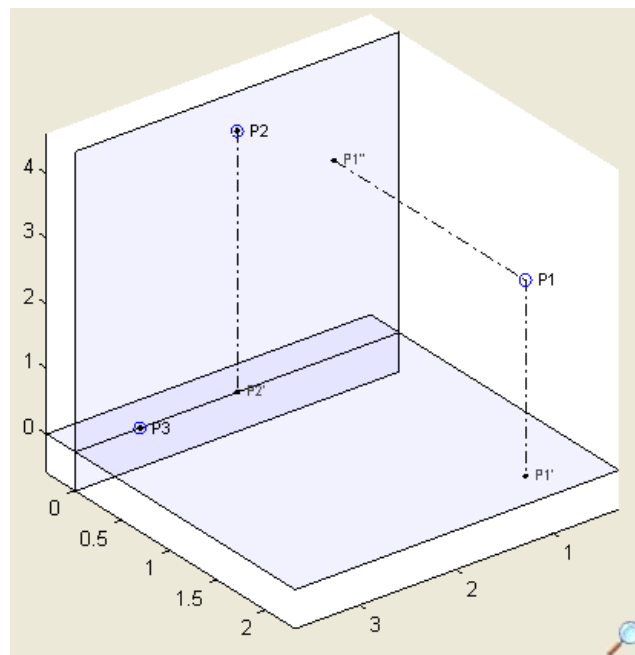
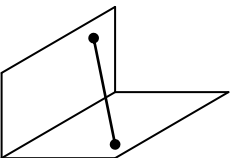
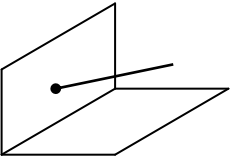
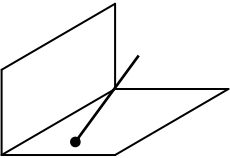
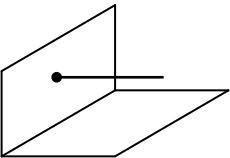
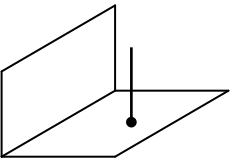
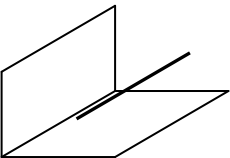
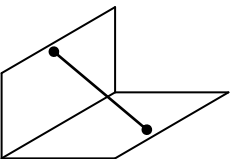


Fig. 7.35 Ejemplo: Puntos 3D

Dibujar Rectas

Para el dibujo de las rectas habrá que tener en cuenta de que tipo se trata, este se refleja en sus características guardadas en el elemento oculto "rectas" como se observa en la Tabla 7.1, como ya se ha mencionado cada recta en "rectas" ocupa seis posiciones, tres para el punto y 3 para el vector director.

Tabla 7.1 Recta vs elemento oculto "rectas"

TIPO RECTA	DIBUJO	RELACIONES
Oblicua		$[P_x P_y P_z u_x u_y u_z]$
Horizontal		$[P_x P_y P_z u_x u_y 0]$
Frontal		$[P_x P_y P_z u_x 0 u_z]$
De Punta		$[P_x P_y P_z 0 u_y 0]$
Vertical		$[P_x P_y P_z 0 0 u_z]$
Paralela a LT		$[P_x P_y P_z u_x 0 0]$
Perfil		$[P_x P_y P_z 0 u_y u_z]$

- *Rectas en dos dimensiones*

Los pasos a seguir para la representación de las rectas en dos dimensiones aparecen en la Fig. 7.36, dependiendo el tipo de recta los caminos a seguir son distintos, debido a sus particularidades.

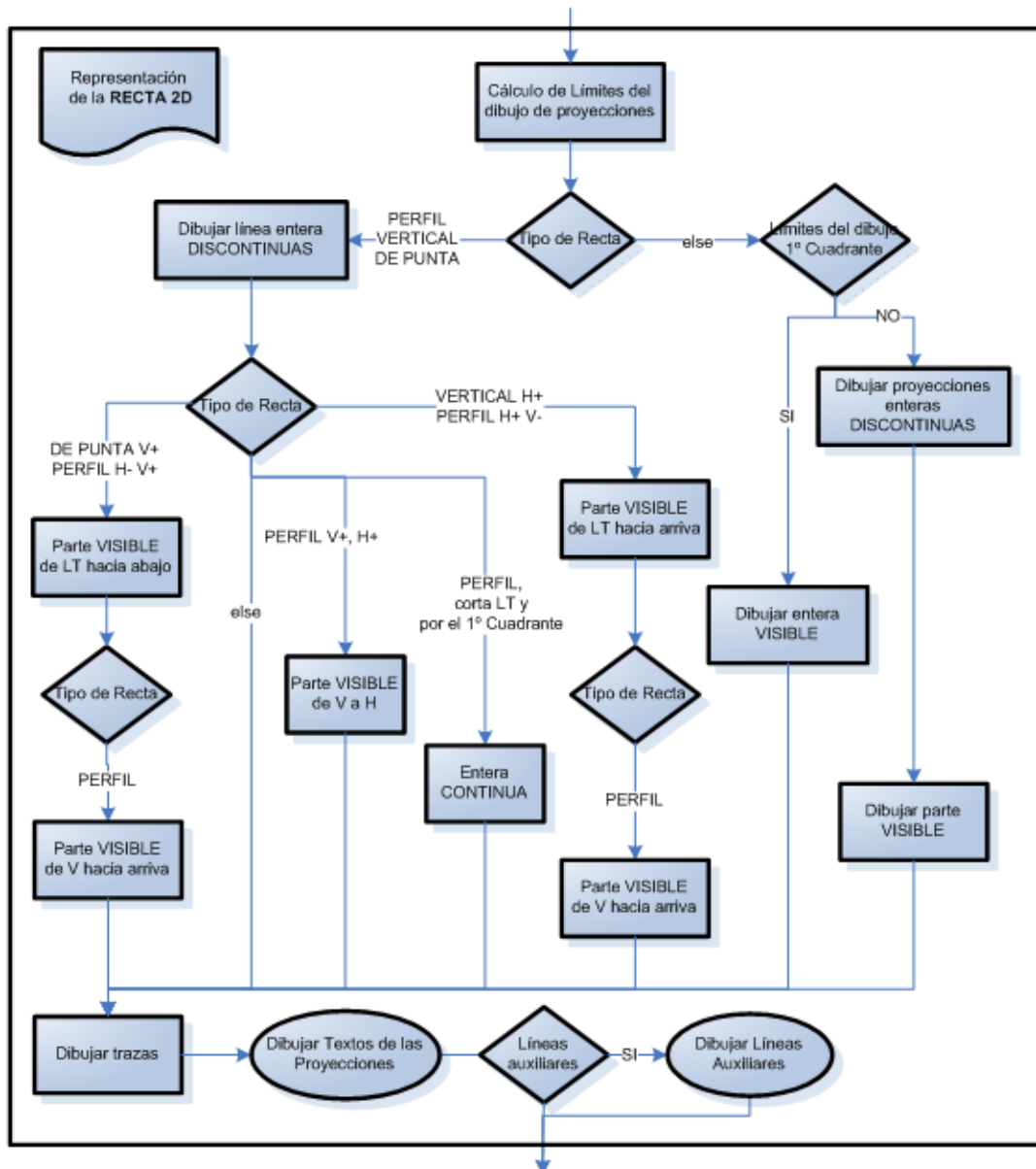


Fig. 7.36 Diagrama de Dibujar: Rectas en 2D

Líneas Fijas:

- **Proyecciones de la recta**

En el diagrama de la Fig. 7.36 se estudia detalladamente los pasos del dibujo de las proyecciones de la recta en dos dimensiones, para los diferentes tipos de rectas. Lo primero es calcular los límites para dibujar cada una de las proyecciones.

Las rectas tienen dos proyecciones, y se representan con línea continua para la zona de la recta que está en el primer cuadrante (parte visible de la recta) y línea

discontinua para el resto de la recta. Para la representación de la recta en dos dimensiones, hay que tener en cuenta los tipos de rectas de la siguiente forma:

- Rectas DE PUNTA, VERTICAL Y DE PERFIL → UNA LÍNEA

En el caso de las rectas de punta y vertical, una de las proyecciones será un punto y en la recta de perfil se superponen sus dos proyecciones, además, para los tres tipos de rectas la coordenada X es constante, por lo que en estos casos sólo se dibuja una línea perpendicular a la Línea de Tierra, con unos tramos discontinuos y otros continuos, dependiendo la posición de la recta.

Como muestra el diagrama de la Fig. 7.36 primero se dibujará una línea entera discontinua y luego dependiendo del tipo y de la posición en el espacio de la recta se superponen los trazos continuos.

En la Fig. 7.37 se representan con sus posibles posiciones en el espacio las rectas verticales y de punta y en la Fig. 7.38 algunas de las posibles posiciones de la recta de perfil:

- Para las rectas verticales, r_3 y r_4 en la Fig. 7.37, la proyección horizontal es un punto que coincide con la traza horizontal. Para el caso de la recta de punta, r_1 y r_2 de la Fig. 7.37, la proyección vertical es un punto coincidente con la traza vertical. Este tipo de rectas sólo tienen una traza, es decir, pasan por dos diedros, ya que las trazas implican cambio de cuadrante, por lo que si pasa por el primer cuadrante la proyección correspondiente a la recta será la mitad visible y mitad oculta (visible de de LT hacia abajo en caso de recta de punta, r_2 , y de LT hacia arriba en caso de vertical, r_3).
- En las rectas de perfil, las dos proyecciones se superponen, por lo que puede haber dos zonas visibles en la misma línea, así dependiendo de la situación de las trazas, puede haber distintas partes visibles como se muestra en la Fig. 7.38. Por ejemplo la recta r_2 , es una recta de perfil que corta en la Línea de Tierra y atraviesa el primer cuadrante por lo que su proyección horizontal es continua de la Línea de Tierra hacia abajo y la vertical de la Línea de Tierra hacia arriba.

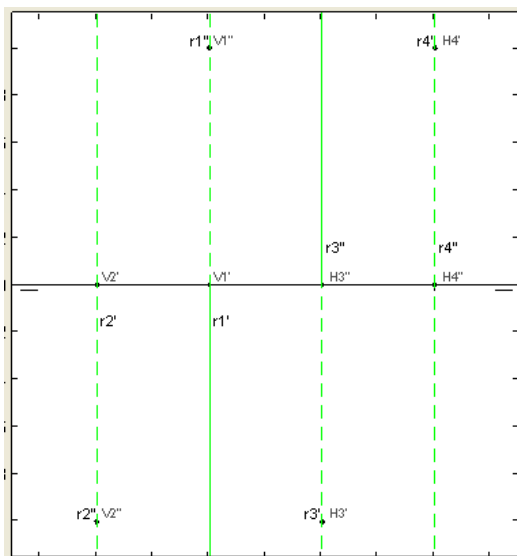


Fig. 7.37 Ejemplo: Rectar Vertical y de Punta 2D

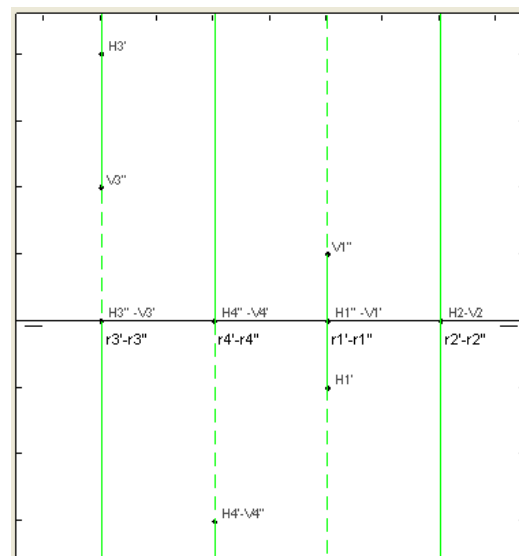


Fig. 7.38 Ejemplo: Rectas de Perfil 2D

- **Rectas con UNA LÍNEA POR PROYECCIÓN**

Lo primero a comprobar es si entre los límites calculados la recta está en la zona visible si fuese así se dibujan las dos proyecciones en continua como la recta r_2 de la Fig. 7.39, en el caso contrario se dibuja las dos proyecciones con trazo discontinuo, y se llama a la función “lineaContinua”, que se explicará a continuación, esta función se encarga de comparar los puntos notables de la recta, es decir, los límites a dibujar y cada una de sus trazas, y a partir de ellos dibuja la parte continua de las proyecciones si la hubiese. En las Fig. 7.39 y Fig. 7.40 hay ejemplos del resultado.

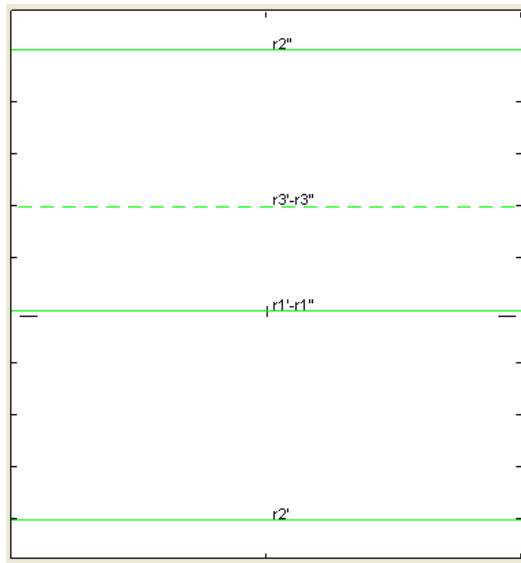


Fig. 7.39 Ejemplo; Recta Paralela a LT 2D

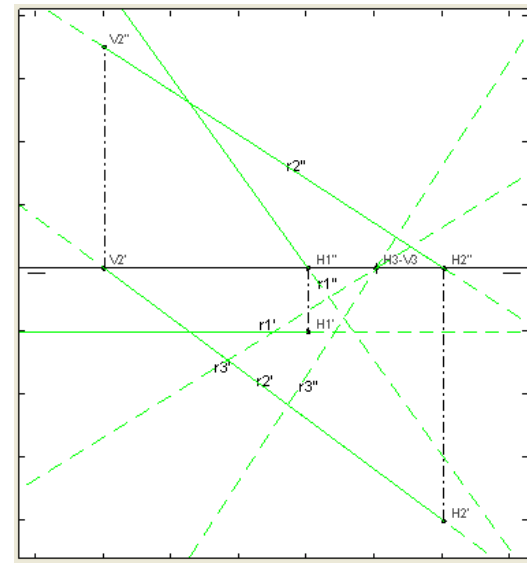


Fig. 7.40 Ejemplo: Rectas 2D

- **Texto de las Proyecciones** (si se selecciona)

Ante el problema de apilotonamiento de textos, se ha buscado que para cada elemento por separado sus textos no se superpongan, por eso se hacen pequeñas validaciones según los tipos de rectas y la posición de las trazas. Los textos de las proyecciones se colocan, por norma general, en el centro de la línea que representa la proyección. Algunos casos a tener en cuenta son las rectas de perfil que como se refleja en la Fig. 7.38 las proyecciones se superponen, al igual que para algunos casos de las rectas paralelas a la Línea de Tierra, como las rectas r_1 y r_3 de la Fig. 7.39. Por otra parte en las rectas de punta y verticales, como las representadas en la Fig. 7.37, una de sus proyecciones coincide con su traza. También se tiene en cuenta que para las rectas que cortan en la Línea de Tierra se deben separar los nombres de las proyecciones de esta, para evitar el apilotonamiento, como pasa con la recta r_3 de la Fig. 7.40.

- **Trazas de la Recta**

Las trazas de una recta, son los puntos que cortan con los planos de proyección, y como cualquier punto, su representación en dos dimensiones consiste en un punto en cada una de sus proyecciones en los planos de coordenados.

Líneas auxiliares. Se dibujan cuando se viene desde la función genérica Dibujar, y se selecciona la representación con líneas auxiliares. Además de lo mencionado sin líneas auxiliares, se dibujan:

- **Línea que une las Proyecciones de las Trazas**

Al estar las trazas en los planos coordenados se unirá: la proyección vertical de la traza vertical con la Línea de Tierra así como la proyección horizontal de la traza horizontal. Se tiene en cuenta que no haya ninguna recta que coincida con la línea, que las coordenadas de las trazas formen parte de las escalas y la posibilidad de la no existencia de alguna o ambas trazas.

- **Textos de las Trazas** (si se selecciona)

Para la creación de los textos de las trazas se tiene en cuenta que la las rectas vertical y frontal no tiene proyección vertical y para las rectas de punta y horizontal no tienen horizontal. Además si la recta corta a la Línea de Tierra sus trazas coincidirán, y para las rectas de perfil siempre coinciden la proyección vertical de la traza horizontal y la horizontal de la traza vertical.

- *Rectas en tres dimensiones*

Líneas Fijas:

- **Recta**

Para dibujar la recta se calculan sus límites en nuestro dibujo, estos límites pueden superar las escalas, de hecho se elige los límites que hagan el segmento dibujado lo más largo posibles. Una vez tenemos los límites comprobamos la visibilidad del segmento, si todo el segmento es visible se dibuja la línea continua, en caso contrario se dibuja entera en trazo discontinuo y se llama a la función "líneaContinua", que se explicará a continuación, esta función se encarga de comparar los puntos notables de la recta, es decir, los límites a dibujar y cada una de sus trazas, y a partir de ellos dibujar la parte continua de la recta si la hubiese.

- **Trazas de la Recta**

- **Texto de la Recta** (si se selecciona)

El texto de la recta se coloca en la parte central del segmento a dibujar de la recta, se tendrá en cuenta que los límites del segmento de la recta pueden superar las escalas, de tal forma que se evitará colocar el texto fuera de ellas.

Líneas auxiliares. Se dibujan cuando se viene desde la función genérica Dibujar, y se selecciona la representación con líneas auxiliares. Además de lo mencionado sin líneas auxiliares, se dibujan:

- **Proyecciones de la Recta**

Los límites de las proyecciones de la recta deben coincidir exactamente con los planos de proyección, así que a partir de los límites del dibujo de la recta se calcula los límites de cada una de sus proyecciones, y seguidamente se dibujarán las proyecciones

usando por cada una de ellas la función “líneaContinua”, es decir, en caso de no ser el segmento dibujado de la parte visible entero, se dibuja la líneas en trazo discontinuo y luego la parte continua a partir de la función. En la Fig. 7.41 se presentan algunos ejemplos de las rectas en tres dimensiones con líneas auxiliares. No se dibujan las proyecciones que coincidan con la misma recta o con cualquier otra, como r2.

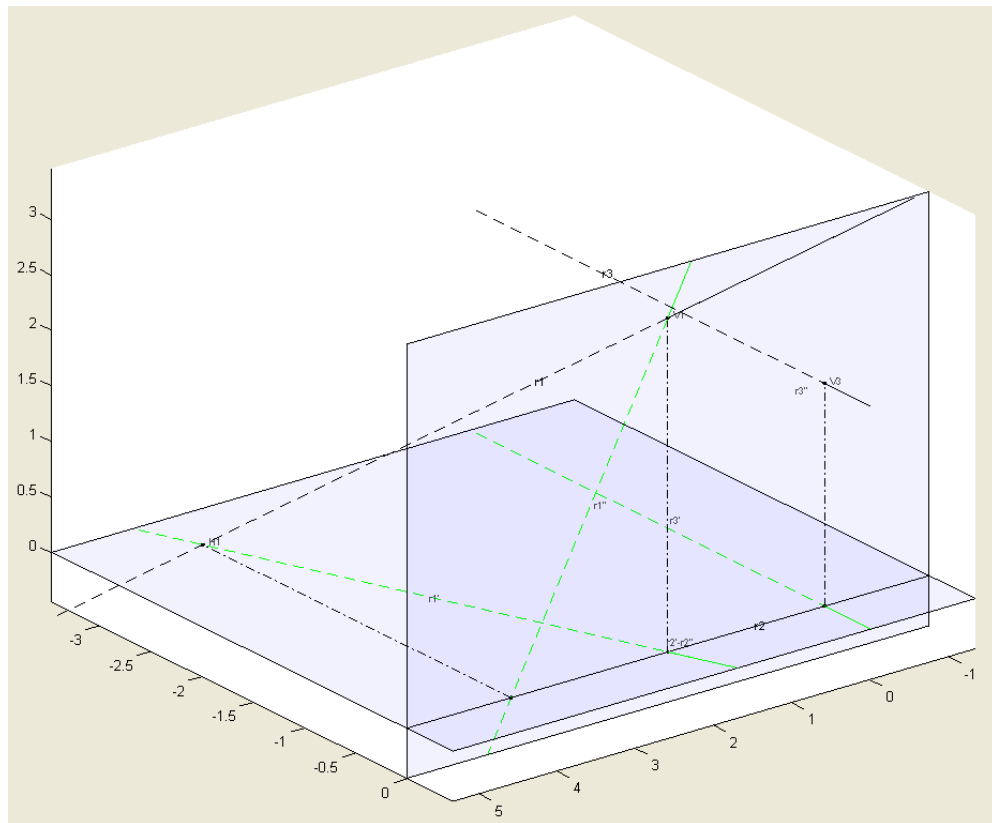


Fig. 7.41 Ejemplo: Rectas 3D

- **Textos de las Proyecciones** (si se selecciona)
Los textos de las proyecciones se colocan el centro de ellas, teniendo en cuenta las que para la recta vertical y de punta una de las proyecciones consiste en un punto (recta r3 de la Fig. 7.41), o que para la recta que coincide con la Línea de tierra tiene proyecciones coincidentes (recta r2 de la Fig. 7.41).
- **Textos de las trazas** (si se selecciona)
Los textos se dibujarán en el caso de que las coordenadas de las trazas entren dentro de las escalas. Para los textos de las trazas se tiene en cuenta que algunas rectas sólo tienen una traza, que para las rectas que cortan en la Línea de Tierra las trazas coinciden.
- **Línea que une la Proyecciones de la Trazas**

- *Función “lineaContinua”*

La función “lineaContinua” es un complemento de la función principal Dibujar para el dibujo de rectas, en concreto de la parte visible, es decir, el segmento en continuo de algunos tipos rectas y sus proyecciones.

Su objetivo, como ya se ha mencionado, es dibujar la parte continua de las rectas y/o sus proyecciones, a partir de puntos clave, que son las trazas de la recta, tanto la vertical como la horizontal, y los límites del dibujo de la recta.

La inicialización de esta función es: lineaContinua(Prtas,ind,dib), donde solamente hay variables de entrada, la salida es el dibujo de la línea continua:

VARIABLES DE ENTRADA

- Prtas: Matriz numérica que contiene las coordenadas de los puntos principales de la recta, el primer y último punto son los que limitan el dibujo de la recta.
- ind: Número que señala la coordenada a partir de la que se ordena la recta.
- dib: Carácter utilizado como indicador del tipo de dibujo, puede ser ‘2D’ que es el dibujo de las proyecciones en dos dimensiones, ‘3D’ el dibujo de la recta en tres dimensiones, ‘y’ o ‘z’ que indica el dibujo de las proyecciones horizontal o vertical en tres dimensiones.

Es un complemento de la función principal Dibujar, se llama para el dibujo de rectas y sus proyecciones.

- Para dibujar el segmento visible de las proyecciones de las rectas Oblicuas, Horizontales y Frontales en dos dimensiones (corten o no a la Línea de Tierra) se le pasa ind=1, es decir coordenada x y dib=2D.
- En la representación en tres dimensiones, se utiliza para dibujar la parte visible de la recta, y de cada una de sus proyecciones, pasándole ind=2 si la recta es de perfil, de punta o vertical e ind=1 para los demás casos, para la recta se le pasa dib=3D, la proyección horizontal dib=z, y la proyección vertical div=y.

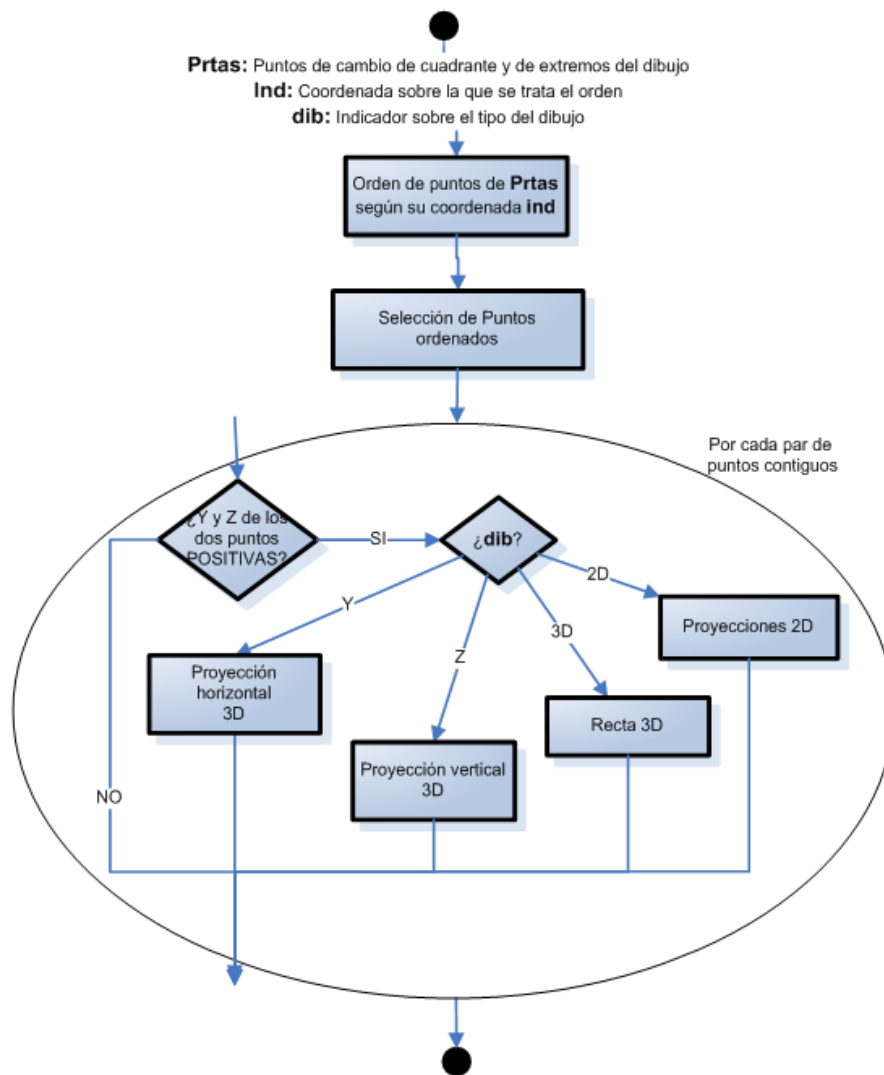


Fig. 7.42 Diagrama Función líneaContinua

El array de entrada Prtas debe contener los puntos notables de la recta, estos son el punto máximo y mínimo que entran en el dibujo de la recta, y los puntos de cambio de diedro. Como muestra la Fig. 7.42 el primero paso será ordenar los puntos dentro de la recta, para eso se ordena según la coordenada indicada en la variable de entrada ind, que ya habrá tenido en cuenta las restricciones, por ejemplo, todos los puntos de una recta de perfil tienen la misma x por lo que no se puede ordenar por esta coordenada.

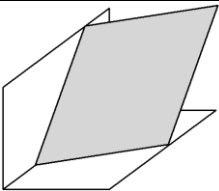
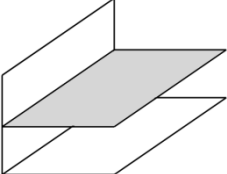
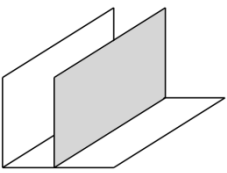
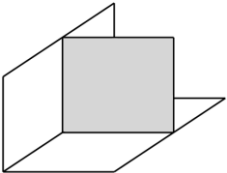
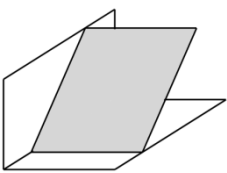
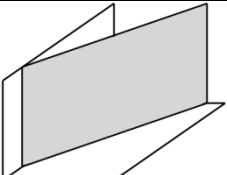
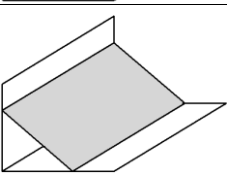
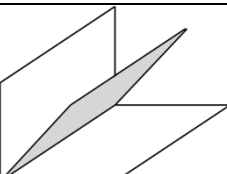
Con el array ordenado sólo se evaluarán los puntos que estén entre el máximo y el mínimo del dibujo, dibujando la línea continua del par de puntos contiguo cuyas Y's y Z's sean positivos (el cero es considerado positivo).

La línea continua dibujada dependerá de la variable de entrada dib, en caso de "2D" se dibujan las dos proyecciones en dos dimensiones, para "3D" la recta en tres dimensiones, para "z" la proyección horizontal en tres dimensiones y para la "y" la proyección vertical en tres dimensiones.

Dibujar Planos

Para el dibujo de los planos habrá que tener en cuenta de que tipo se trata, este se refleja en sus características guardadas en el elemento oculto “planos” como se observa en la Tabla 7.2, como ya se ha mencionado cada plano en “planos” ocupa cuatro posiciones, una por cada constante de la ecuación paramétrica.

Tabla 7.2 Planos vs elemento oculto "planos"

TIPO PLANO	DIBUJO	RELACIONES
Oblicuo		$[A\ B\ C\ D]$
Horizontal		$[0\ 0\ C\ D]$
Frontal		$[0\ B\ 0\ D]$
Perfil		$[A\ 0\ 0\ D]$
Canto		$[A\ 0\ C\ D]$
Vertical		$[A\ B\ 0\ D]$
Paralelo a LT		$[0\ B\ C\ D]$
Pasa por LT		$[0\ B\ C\ 0]$

- *Planos en dos dimensiones*

Líneas Fijas:

- **Trazas del Plano**

Con la función “*limentesPlano*” se consiguen los puntos limites del dibujo de las trazas según las escalas del dibujo y, en caso de tener, el punto de corte con la Línea de Tierra. Se dibuja la parte visible de las trazas con línea continua como en el plano a1 de la Fig. 7.43, en caso de no haber parte visible se dibuja toda la traza en trazo discontinuo. En la Fig. 7.43 aparecen algunos ejemplos de la representación de planos en dos dimensiones, entre ellos el plano a2, que pasa por la Línea de Tierra y necesita un punto extra para su correcta representación, este punto se dibuja en junto con los demás puntos.

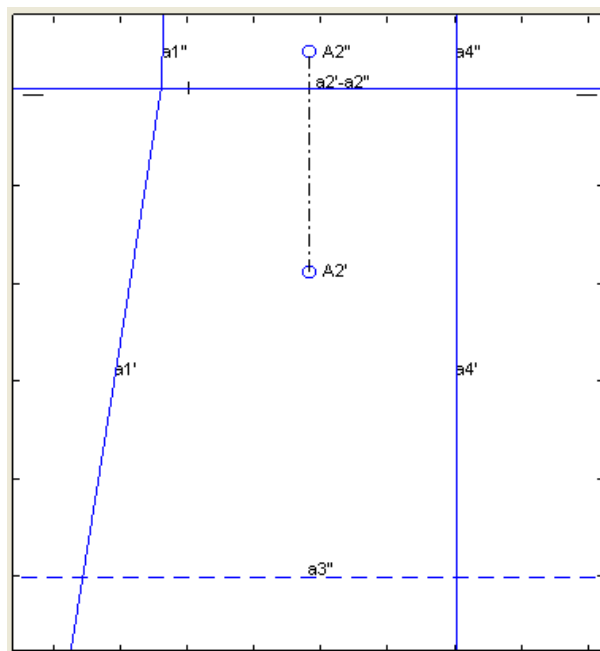


Fig. 7.43 Ejemplo: Planos 2D

- **Textos de las Trazas** (si se selecciona)

Se colocan en el centro del dibujo de la traza. Habrá que tener en cuenta que hay planos con una sola traza como el a3 de la Fig. 7.43, que para los planos que pasan por la Línea de Tierra como el a2 sus trazas son coincidentes y además tienen un punto que identificar que se habrá representado junto con los puntos a dibujar.

- *Planos en tres dimensiones*

Para el dibujo de planos en tres dimensiones utilizaremos la función “*lineasContinuas*”, con la que se obtendrán los puntos de los extremos de las trazas que están dentro de las escalas, como se observa en la Fig. 7.44.

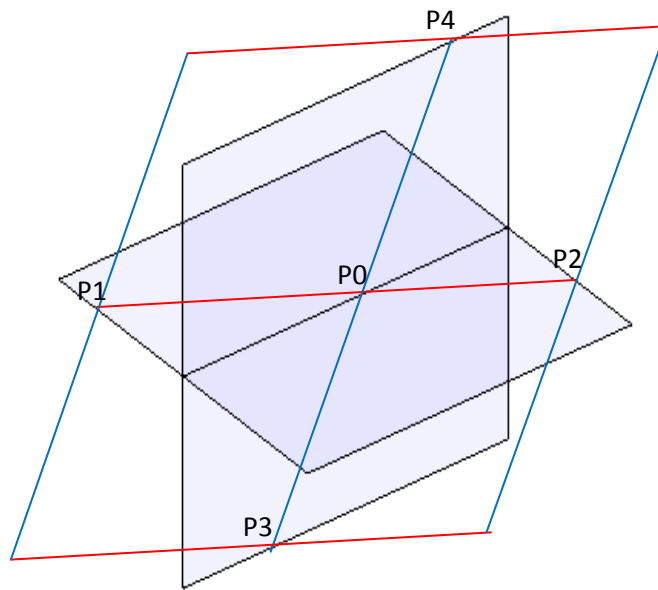


Fig. 7.44 Dibujo de Planos en 3D

Líneas Fijas:

- **Trazas del Plano**

Una vez se obtienen los puntos que limitan las trazas, se unen como muestra la Fig. 7.44.

- **Plano**

Para dibujar el plano, se trazan paralelas de cada una de las trazas por los puntos que limitan la otra traza como muestra la Fig. 7.44. Para planos con trazas paralelas, coincidentes o con una sola traza se tendrán en cuenta las escalas para calcular los límites del plano. En la Fig. 7.45 se muestran algunos ejemplos de planos con líneas auxiliares.

- **Textos del Plano** (si se selecciona)

El texto identificativo de un plano irá en una de las esquinas del plano que estén dentro de las escalas.

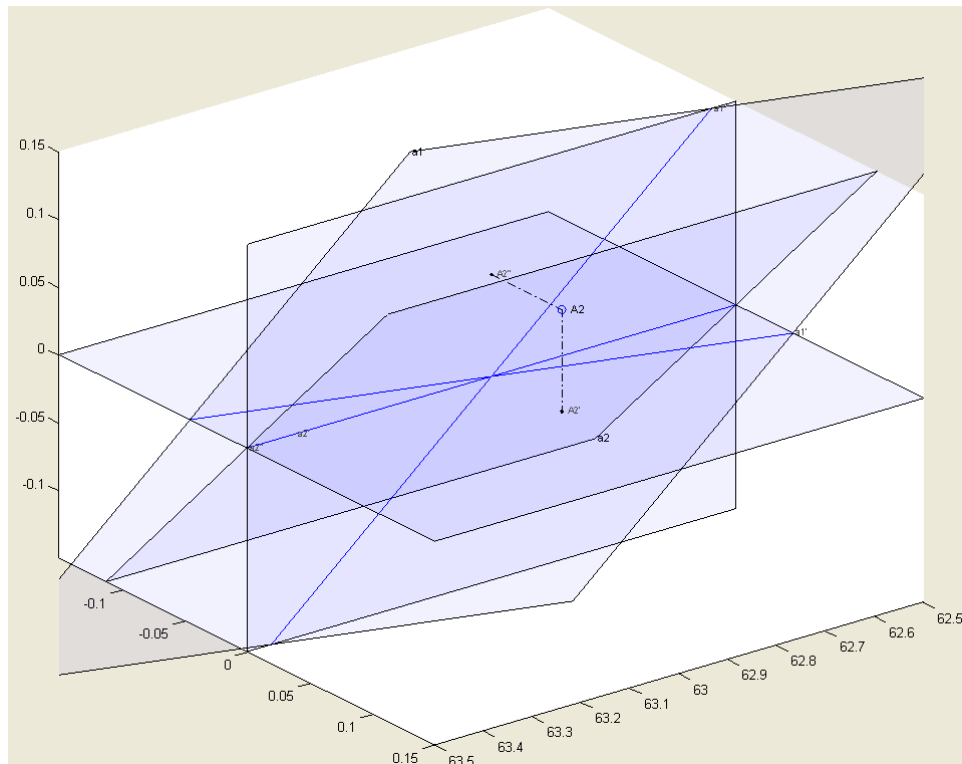


Fig. 7.45 Dibujo de Planos en 3D

Líneas auxiliares. Se dibujan cuando se viene desde la función genérica Dibujar, y se selecciona la representación con líneas auxiliares. Además de lo mencionado sin líneas auxiliares, se dibujan:

- **Textos de las Trazas** (si se selecciona)

Los textos de las trazas se dibujan en los extremos de de estas, teniendo en cuenta que para los planos que pasan por la Línea de Tierra se colocan los textos uno al lado del otro, como en el plano a1 de la Fig. 7.45.

- *Función “limitesPlano”*

La función “limitesPlano” es un complemento de la función principal Dibujar para el dibujo de planos, que calcula los límites de las trazas del plano que a su vez se utilizan para limitar el dibujo del plano.

Su objetivo, como se ya se ha mencionado, es calcular los límites de las trazas del plano a partir de las escalas, de tal forma que cada uno de los puntos límite de las rectas de las trazas deben pertenecer a una de las “rectas” límites de los dibujos de los planos de proyección, que a su vez dependen de las escalas. Además se calcula el punto de corte con la Línea de Tierra.

La inicialización de esta función es: $[P0, trazas] = \text{limitesPlano}(A, B, C, D, \text{lim})$, donde se puede observar que tiene dos variables de salida y cuatro variables de entrada:

VARIABLES DE ENTRADA

A,B,C,D: Variables numéricas que forman la ecuación del plano $Ax+By+Cz+D=0$.

lim: Vector numérico de seis posiciones con las escalas.

VARIABLES DE SALIDA

P0: Vector numérico de tres posiciones con las coordenadas del punto de corte con Línea de Tierra.

trazas: Matriz numérica de cuatro filas y tres columnas, las dos primeras filas corresponden los dos puntos límites de la traza horizontal y las dos últimas a los de la traza vertical.

Es un complemento de la función principal Dibujar y se llama en tres ocasiones:

- Para recalcular las escalas.
- Para el dibujo de las trazas en dos dimensiones.
- Para el dibujo del plano y sus trazas en tres dimensiones.

En la Fig. 7.46 se observa el cálculo de los límites del dibujo de una traza horizontal cualquiera, para ello se tiene en cuenta que las escalas que la limitan pueden ser la X o la Y, con estas escalas se forma el cuadrado que representa el Plano de proyección Horizontal, siendo los límites del dibujo de la traza el corte de ésta con el cuadrado. Para ello se calcula primero cada uno de los límites con respecto a una escala, se comprueba si están dentro del cuadrado, en caso de que no se cumpla en alguno de los límites, se recalcula con la otra escala.

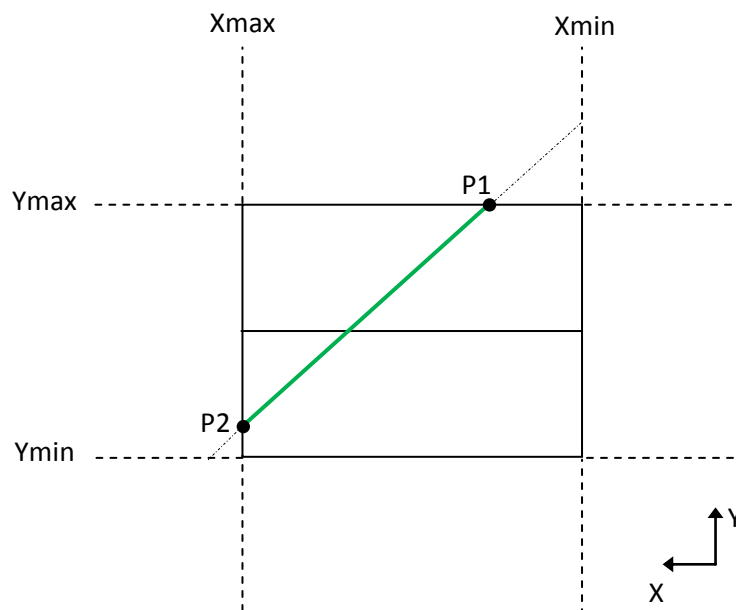


Fig. 7.46 Trazas en el plano Horizontal

En la Fig. 7.47 se observa el cálculo de los límites del dibujo de una traza vertical, que sigue el mismo procedimiento que para la horizontal pero con las escalas X y Z formando el cuadrado que representa el Plano de proyección Vertical, siendo los límites del dibujo de la traza el corte de ésta con el cuadrado. Para ello se calcula primero cada uno de los límites con respecto a una escala, se comprueba si están dentro del cuadrado, en caso de que no se cumpla en alguno de los límites, se recalcula con la otra escala.

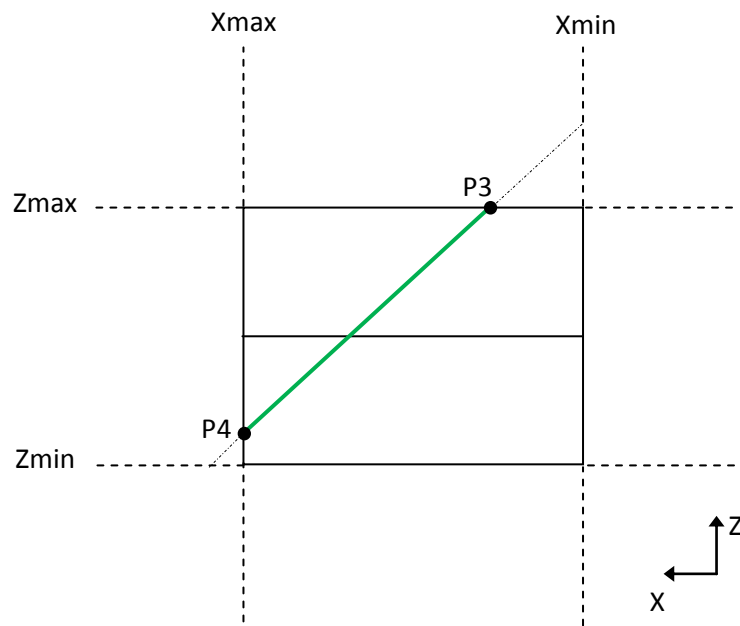


Fig. 7.47 Trazas en el plano Vertical

Una vez calculado los límites se guardan en el array de salida $trazas = [P1; P2; P3; P4]$.

Para el cálculo del punto de corte con la Línea de Tierra, se tiene en cuenta que el punto será de la forma $P(x \ 0 \ 0)$, sustituyendo en la ecuación paramétrica del plano $Ax + B0 + C0 + D = 0$, de donde sacamos $x = -D/A$.

En esta función se tendrá en cuenta los planos con condiciones especiales que como se refleja en la Tabla 7.2 se reconocen a partir de su ecuación paramétrica:

En el caso de que $A = 0$, no hay punto de corte con la Línea de Tierra

El plano horizontal sólo tiene traza vertical.

El plano vertical sólo tiene traza horizontal.

7.3.5. Funciones Auxiliares Generales

Sacar Elementos de una Cadena: Función `sacaIndices`

A la función “*sacaIndices*” se la llama varias veces a lo largo del programa para sacar de una cadena de caracteres los elementos introducidos.

Los objetivos de esta función son recuperar, a partir de la primera posición del nombre de los elementos:

- El índice, o parte numérica del nombre, del elemento.
- Las posiciones que ocupa dentro de la cadena de caracteres introducida.

La entrada a esta función depende de datos introducidos por el usuario, por lo que puede encontrar inconsistencias con lo que su salida debe ser un mensaje de aviso.

La inicialización de esta función es: $[elementos, intervalo, txt] = sacaIndices(string, inicio)$, donde se puede observar que tiene dos variables de entrada y tres variables de salida:

VARIABLES DE ENTRADA

string: Cadena de caracteres de donde se recuperan los elementos.

inicio: Vector numérico con las posiciones de string a partir de las que se sacarán los elementos.

VARIABLES DE SALIDA

txt: Cadena de caracteres que en caso de error por datos de entrada contendrá el mensaje de error.

elementos: Vector numérico, que regresa los índices de los elementos.

Intervalo: Matriz de números, que devuelve en cada fila las posiciones que ocupa el punto.

Se llama desde la función principal Nuevo:

- Al crear una recta para recuperar los puntos que la forman.
- Al sacar los elementos para crear un plano, una vez para las rectas y otra para los puntos.
- Al crear un plano si se crea a partir de alguna recta, para sacar de la memoria visible los puntos que forman la recta.

Se llama también desde la función *“borrarAuxiliar”*:

- Para recuperar los elementos de la memoria visible que pueden tener que renombrarse.

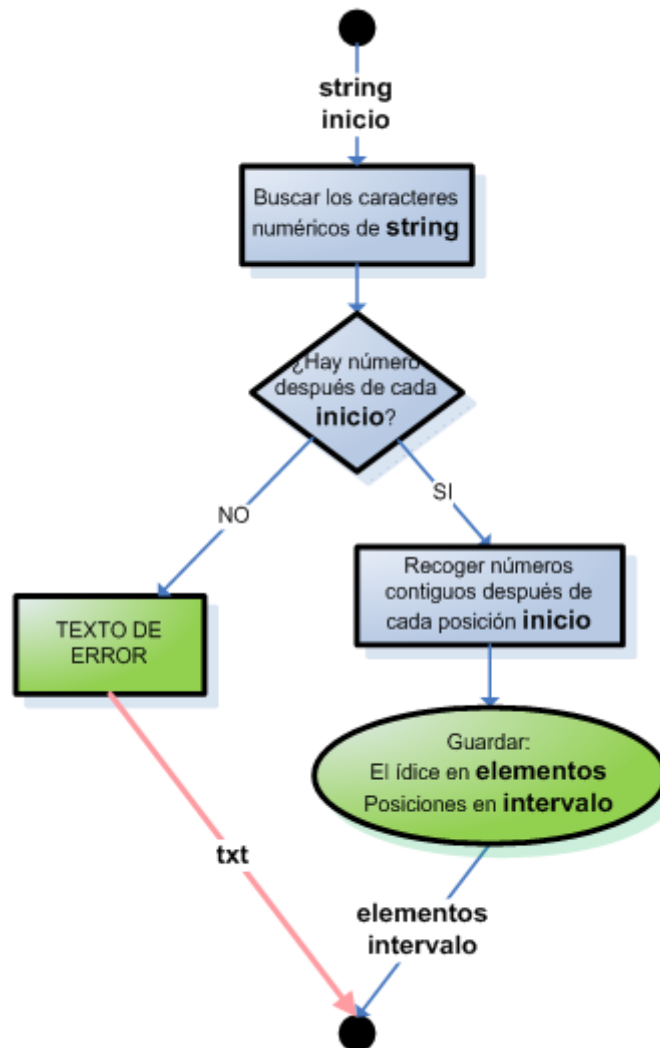


Fig. 7.48 Diagrama de sacaIndices

En la Fig. 7.48 se ven los pasos necesarios para conseguir los objetivos:

1. Lo primero será recoger todos los caracteres numéricos de la string, así como la posición de estos.
2. Si detrás de cada posición de inicio introducida no hay un número significa que uno de los elementos está mal introducido, termina la función enviando un mensaje de error que deberá mostrarse en un aviso.
3. El valor numérico del nombre del elemento lo formarán todos los números de la cadena que haya después e la posición de inicio introducida.
4. Una vez que ya tenemos el valor numérico lo guardamos en las variables de salida, también guardamos las posiciones que ocupa este nombre en la cadena de entrada.

Comparación de Rectas: Función relacionRectas

La función “*relacionRectas*” es llamada desde varias funciones principales para saber la relación que hay entre dos rectas.

El objetivo de esta función es comprobar la relación que existe entre las dos rectas introducidas. Las relaciones posibles serán:

- Rectas iguales.
- Rectas paralelas.
- Rectas que se cortan, en cuyo caso se calcula el punto de corte.

La inicialización de esta función es: [igu,par,cor,corte] = relacionRectas(r1, r2), con cuatro variables de salida y dos variables de entrada:

VARIABLES DE ENTRADA

- r1: Vector numérico de seis posiciones con las características de una de las rectas a comparar, las tres primeras posiciones corresponden al punto y las tres últimas al vector director.
- r2: Vector numérico de seis posiciones con las características de una de las rectas a comparar, las tres primeras posiciones corresponden al punto y las tres últimas al vector director.

VARIABLES DE SALIDA

- igu: Variable numérica que indica igualdad de rectas, para rectas iguales igu=1.
- par: Variable numérica que indica paralelismo de rectas, para rectas paralelas par=1.
- cor: Variable numérica que indica si las rectas cortan, para rectas que cortan cor=1.
- corte: Vector numérico con las coordenadas del punto de corte entre rectas. Para evitar posibles errores causados por el redondeo, se rellena mientras las rectas no son paralelas o iguales.

Se llama desde la función principal Nuevo:

- Al introducir una recta para saber si está repetida.
- Al crear un plano con tres puntos para saber si los tres puntos forman parte de la misma recta.
- Al crear un plano con una recta y un punto para saber si el punto forma parte de la recta.
- Al crear el plano con dos rectas para saber si son paralelas o se cortan.

Se llama también desde la función principal Dibujar:

- Al recalcular las escalas para saber si los puntos del plano se dibujan dentro de los límites del dibujo del plano.
- Al dibujar las proyecciones de la recta, para comprobar si coincide con alguna recta.
- Al dibujar las trazas del plano en tres dimensiones para comprobar si coincide con alguna proyección de alguna recta.

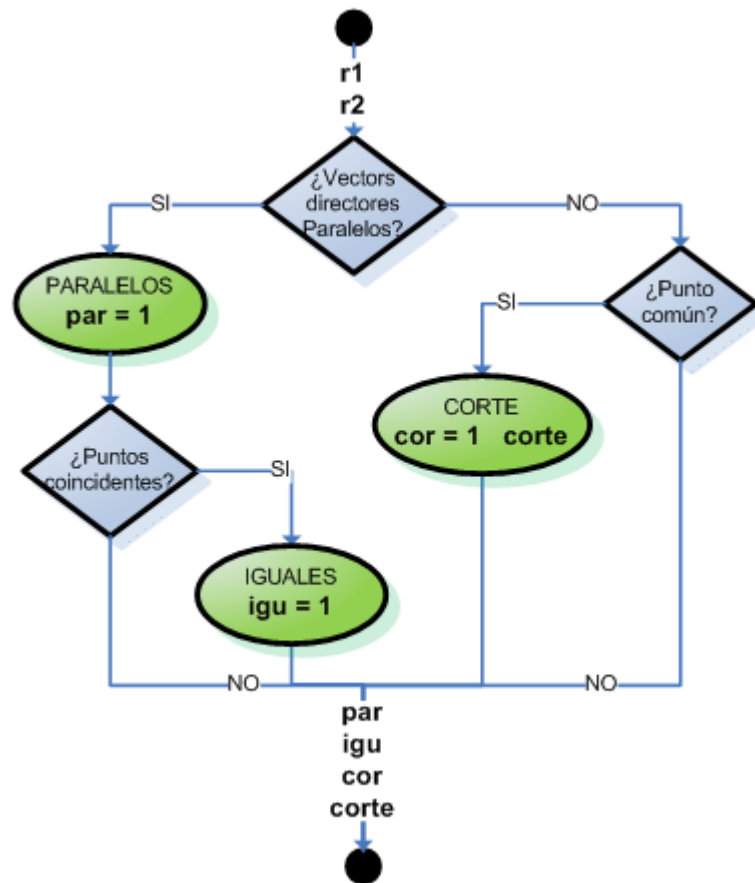


Fig. 7.49 Diagrama de relacionRectas

Como se ve en la Fig. 7.49 para saber que relación tienen las rectas se hacen tres validaciones:

- La primera para saber si son paralelos, es decir, se comprueba si los vectores directores son proporcionales: $\vec{v} = k\vec{u}$, $\rightarrow (v_x, v_y, v_z) = (ku_x, ku_y, ku_z)$
- En el caso de ser paralelas y coincide algún punto serán rectas coincidentes.
- Si coincide algún punto sin ser paralelas, significa que son rectas que se cortan.

TÍTULO IV: CONCLUSIONES Y DESARROLLOS FUTUROS

8. Conclusiones

Se ha conseguido una aplicación de uso muy sencillo e intuitivo, con una funcionalidad mayor de la esperada inicialmente.

El alumno podrá utilizarlo sin ningún tipo de conocimiento de sistema diédrico, y con la selección de elementos le permite jugar con todos ellos para ver los conceptos que irá aprendiendo.

La claridad supera en mucho las expectativas iniciales, con la posibilidad de abrir una ventana independiente con el dibujo más grande y el giro en tres dimensiones. Además de que con la selección de dibujo de más o menos líneas y de textos a ver, se consiguen unas representaciones más claras y menos colapsadas. Otro punto en este sentido, que proporciona una visión más rápida, son los colores utilizados para representación, siendo en dos y tres dimensiones y sin dar un aspecto caótico. Todo esto sin perder el dinamismo y la rapidez de selección y dibujo.

Se ha buscado un código extremadamente ordenado, limpio y estructurado, agrupando dentro de lo posible todos los pasos con el mismo objetivo y comentando el código, de tal forma que cuando otro desarrollador se empiece a cambiar el código para introducir nuevas opciones o mejoras, le sea fácil y rápido localizar las ordenes a modificar.

Se ha utilizado muchas de las grandes ventajas del MATLAB, especialmente gráficas.

9.Desarrollos Futuros

Esta aplicación tiene una posible ampliación futura muy evidente, por ahora solo se permite el dibujo de los elementos más simple, que son el punto, la recta y el plano.

En estos momentos es una aplicación muy útil para posibles alumnos sin ninguna base de sistema diédrico, para la visualización de elementos en sistema diédrico, que con tantas dificultades se presenta inicialmente, pero una vez ya se tiene conocimientos básicos, queda en un programa excesivamente simple y pierde su valor, ante esto se proponen unas ampliaciones muy útiles para que el alumna pueda seguir mas allá de la toma de contacto:

- Representación de elementos en su tercera proyección.
- Calculo de intersecciones, entre rectas o planos.
- Realización de giros.
- Realización de abatimientos.
- Dibujo de elementos más complicados.
- Representación de superficies y sólidos.

Como se ve, el posible futuro de esta aplicación esta ligado a un extenso temario. De hecho esta aplicación pretende ser el inicio de una mucho más grande que englobe lo máximo posible la representación de figuras y elementos en sistema diédrico, quedando aún mucho camino por delante.

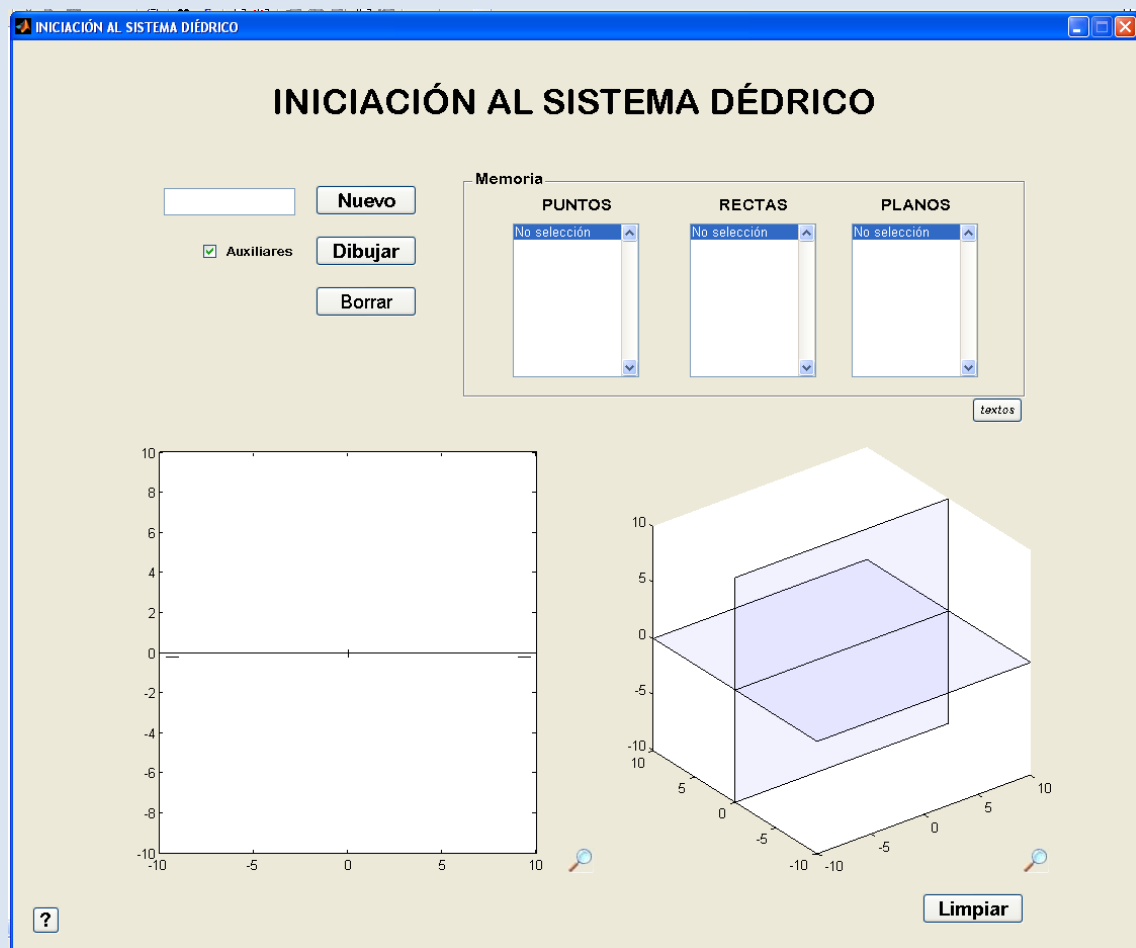
BIBLIOGRAFÍA

- [1] CUADRADO VICENTE, José Antonio. "Laboratorios virtuales para el estudio de la Geometría". Director: CFIE Vitigudino. Instituto de Tecnologías Educativas, 2008.
- [2] PÉREZ MORALES, Máximo Román. "Integración de las TIC al proceso de enseñanza aprendizaje de la Gráfica de Ingeniería". Universidad Central "Marta Abreu" de Las Villas, Departamento de Mecánica Teórica y Dibujo, 2008.
- [3] BLANCO CABALLERO, M., MARTÍN PANERO, A), PRÁDANOS DEL PICO, R., RODRÍGUEZ OVEJERO, Q., SANZ ARRANZ, J., PARRA GONZALO, E. , SAN MARTÍN OJEDA, M., SERRANO SANZ, J. "Estudio del Sistema Diédrico Mediante un Tutorial Multimedia". Universidad de Valladolid.
- [4] ATENCIA, Javier, NESTAR, Raúl. *Aprenda Matlab 6.0 como si estuviera en primero*. Colección: "Aprenda ..., como si estuviera en primero", Universidad de Navarra, TECNUN, 2001.
- [5] 984-2010- The MathWorks, Inc. Página oficial de MATALAB, ayuda. Disponible a 1 de Abril de 2010 en:
<http://www.mathworks.com/access/helpdesk/help/techdoc/>
- [6] Todo dibujo, Technology BBI digital. Disponible a 1 de Abril de 2010 en:
http://www.tododibujo.com/index.php?cPath=298&main_page=site_map
- [7] Sistema Diédrico 2º Bachillerato, 2005. Disponible a 1 de Abril de 2010 en:
<http://www.miajas.com/DibujoTec/defplano.htm>
- [8] Educared, Wilillerato, Dibujo Técnico. Disponible a 1 de Abril de 2010 en:
[http://portales.educared.net/wikillerato/Dibujo Tecnico](http://portales.educared.net/wikillerato/Dibujo_Tecnico)
- [9] Educared, Wilillerato, Matemáticas. Disponible a 1 de Abril de 2010 en
<http://portales.educared.net/wikillerato/Matematicas>

ANEXOS

Manual de Usuario de la Aplicación:

INICIACIÓN AL SISTEMA DIÉDRICO



Universidad Carlos III de Madrid

Departamento de Ingeniería Mecánica

Isabel Roa Prieto

Índice

Introducción	1
Instalación	1
Componentes de la aplicación	1
Nomenclatura utilizada y Elementos	3
Instrucciones de Uso	5
Crear elementos.....	5
Dibujar	6
Borrar	7
Limpiar.....	8
Ayuda	8

Índice de ilustraciones

<i>Ilustración 1 Componentes de la aplicación</i>	<i>2</i>
<i>Ilustración 2 Nomenclatura del Punto</i>	<i>3</i>
<i>Ilustración 3 Nomenclatura de la Recta</i>	<i>4</i>
<i>Ilustración 4 Nomenclatura del Plano</i>	<i>5</i>

Introducción

La aplicación “Iniciación al Sistema Diédrico” nace de la necesidad de una herramienta dinámica, sencilla y clara, para ayudar a alumnos de Dibujo Técnico a comprender los conceptos más básicos del Sistema Diédrico, que con tanta dificultad se entienden. Para ello se permite dibujar de forma sencilla e intuitiva los elementos más básicos, es decir, el punto, la recta y el plano en tres dimensiones y su representación en dos dimensiones según los conceptos del Sistema Diédrico.

Una ventaja clarísima de esta aplicación, es la cantidad de posibilidades que tiene para la visualización de los elementos, permitiendo al usuario interactuar con la herramienta, para ver los elementos de la forma más clara según el caso, así se le permite elegir si se quieren dibujar todas las líneas representativas de cada elemento, o sólo las indispensables, elegir los elementos a representar, eliminar los nombres de los componentes de los elementos o abrir ventanas más grandes con los dibujos que en caso de tres dimensiones nos permite girar.

IMPORTANTE: Está aplicación está preparada para trabajar con una configuración de pantalla de 1280 por 1024 pixeles.

Instalación

Para ejecutar el programa es necesario tener instalado el compilador MATLAB(R) Compiler Runtime 7.9, si ya se tiene basta con ejecutar el archivo “SDiedrico.exe” cada vez que se quiera utilizar.

En caso de no tener el compilador se deberá instalar, para ello, ejecutamos el archivo “Diedrico_pkg.exe”, instalando todas las opciones por defecto y en caso de preguntarnos si queremos reemplazar algún fichero contestaremos ‘y’.

Componentes de la aplicación

La Ilustración 1 es una imagen de la aplicación al iniciarse, es decir la ventana general de la aplicación, a la que se llama espacio de trabajo, en ella se enumeran todos los componentes con los que el usuario interactúa:

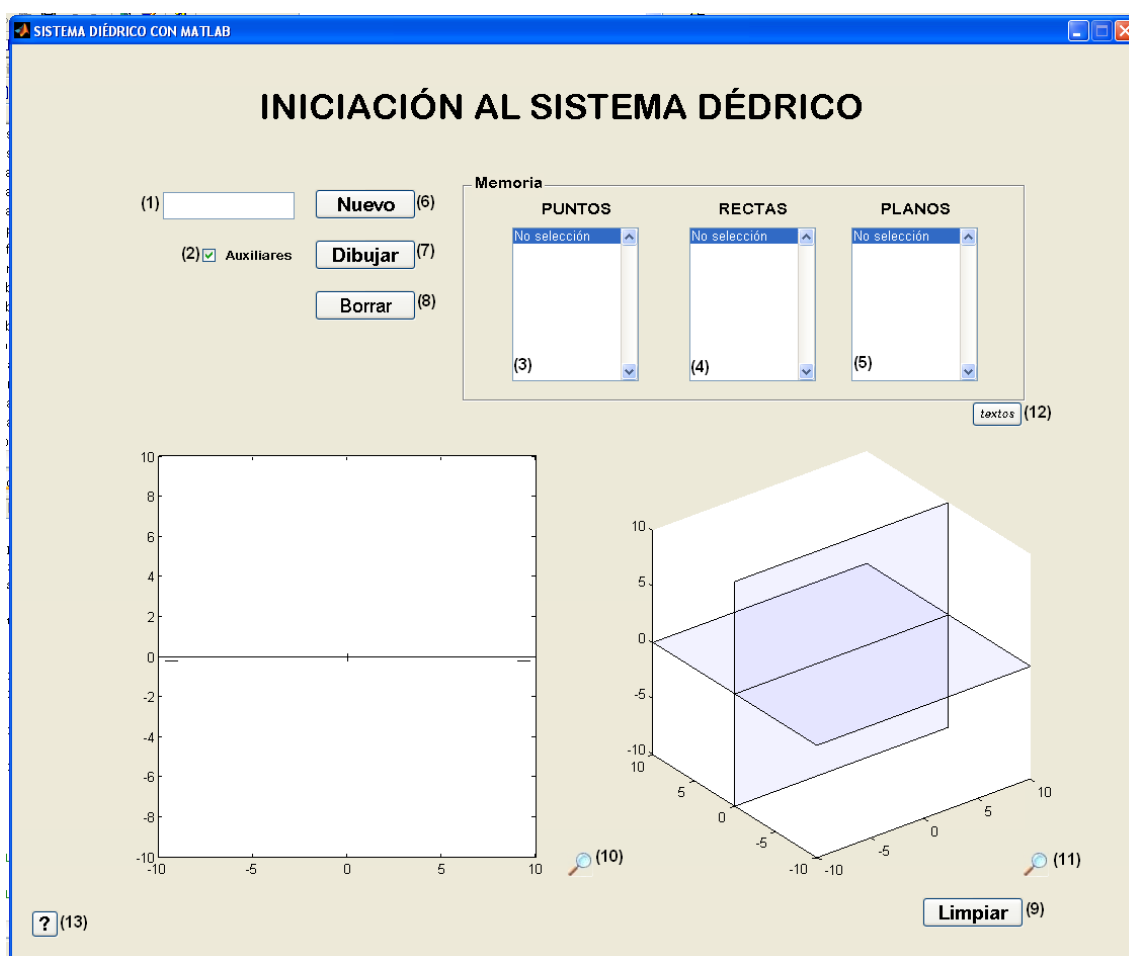


Ilustración 1 Componentes de la aplicación

- (1) **Campo de texto:** Para introducción de datos directamente por el usuario para la creación de elementos.
- (2) **Selección Auxiliar:** Para la selección del dibujo con los mínimos datos posibles.
- (3) **Listado de Puntos:** Para la visión y selección de cada uno de los puntos.
- (4) **Listado de Rectas:** Para la visión y selección de cada una de las rectas.
- (5) **Listado de Planos:** Para la visión y selección de cada uno de los planos.
- (6) **Botón Nuevo:** Para crear elementos nuevos.
- (7) **Botón Dibujar:** Para dibujar elementos.
- (8) **Botón Borrar:** Para borrar elemento.
- (9) **Botón Limpiar:** Para reiniciar el espacio de trabajo.
- (10) **Lupa 2D:** Para dibujar los elementos en una figura grande para dos dimensiones.
- (11) **Lupa 3D:** Para dibujar los elementos en una figura grande para tres dimensiones.
- (12) **Botón Ayuda:** Para abrir un texto de ayuda.

(13) Botón textos: Para quitar o poner todos los textos de los elementos.

Nomenclatura utilizada y Elementos

Puntos

Los puntos se representan con sus dos proyecciones, la proyección horizontal se le llamará con el nombre del punto seguido de (') y la proyección vertical seguido de ("), en la Ilustración 2 hay un ejemplo de la representación de un punto.

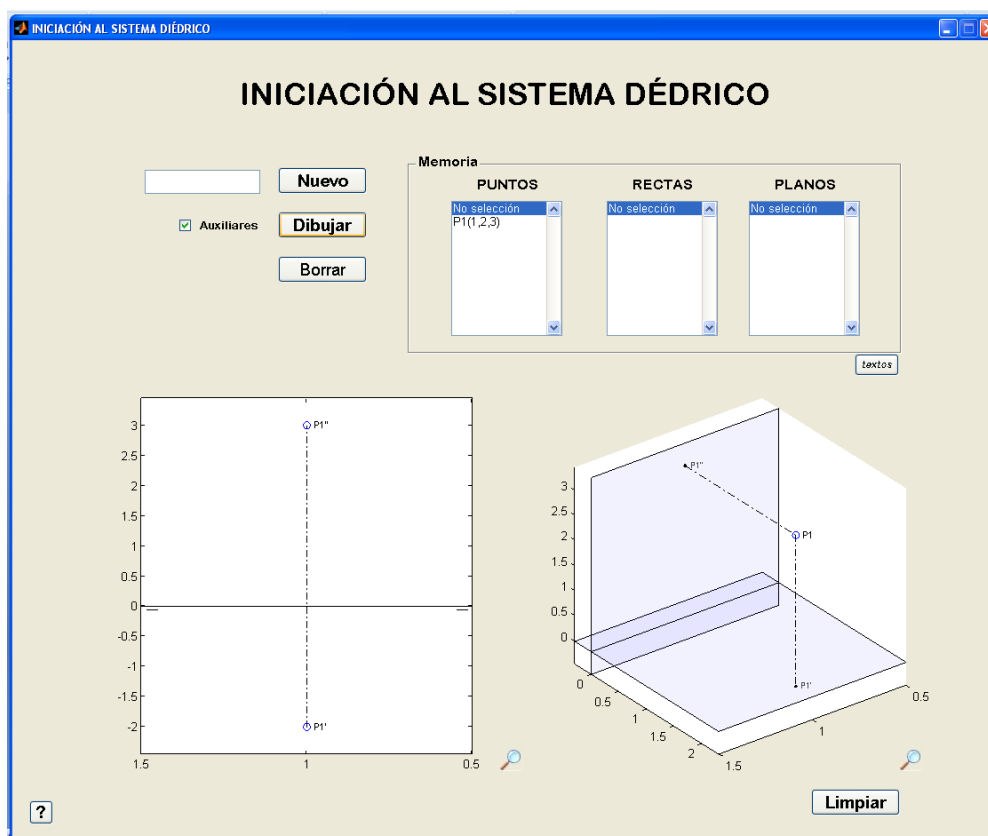


Ilustración 2 Nomenclatura del Punto

La aplicación distingue los datos necesarios y auxiliares para la visualización del punto. En la Ilustración 2 se representan todos los datos, siendo auxiliares las proyecciones del punto en tres dimensiones.

Rectas

Las rectas se representan con sus dos proyecciones y sus trazas, la proyección horizontal se le llamará con el nombre de la recta seguido de (') y la proyección vertical seguido de ("), la traza horizontal se representa con una (H) seguida del número que da nombre a la recta, y sus proyecciones como cualquier otro punto se representa con (') para la horizontal y (") para la vertical, al igual que en la traza vertical que se nombra con una (V) seguida del número que da nombre a la recta. En la Ilustración 3 hay un ejemplo de la representación de una recta.

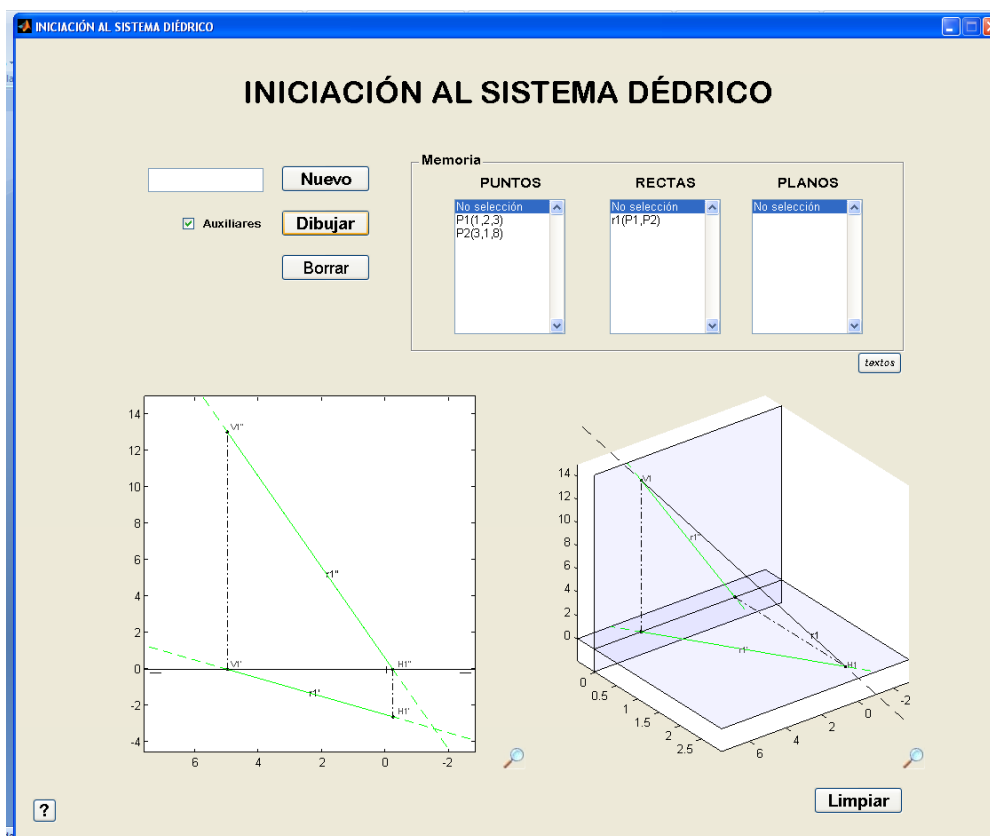


Ilustración 3 Nomenclatura de la Recta

La aplicación distingue los datos necesarios y auxiliares para la visualización de la recta. En la Ilustración 3 se representan todos los datos, siendo auxiliares las proyecciones en tres dimensiones y, los nombres de las trazas tanto en dos como en tres dimensiones.

Planos

Los planos se representan con sus dos trazas, la traza horizontal se le llamará con el nombre de la recta seguido de ('') y la traza vertical seguido de ("). En la hay un ejemplo de la Ilustración 4 representación de un plano.

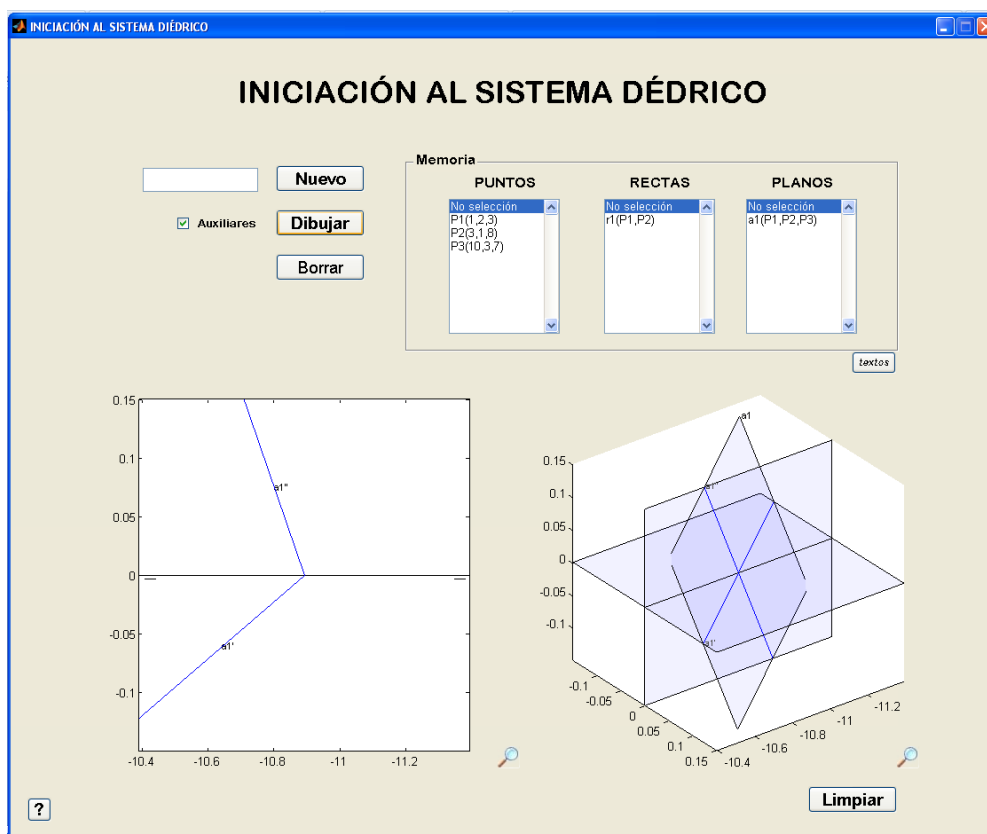


Ilustración 4 Nomenclatura del Plano

La aplicación distingue los datos necesarios y auxiliares para la visualización del plano. En la Ilustración 4 se representan todos los datos, siendo auxiliares los nombres de las trazas en tres dimensiones.

Instrucciones de Uso

- *Crear elementos*

La aplicación Diédrico con MATLAB permite representar los elementos más sencillos del espacio, es decir, puntos, rectas y planos.

Los posibles errores introducidos se avisarán con un mensaje aclarativo.

Los elementos creados se irán nombrando automáticamente, y guardándose en las Listas de elementos, desde donde se comprueba el nombre q recibe.

Los elementos se pueden crear a través del campo de texto (1), o de la selección en los listados de puntos y/o rectas (3), (4). Si hay alguna selección lo tomará como base del nuevo elemento, por delante del contenido del campo de texto (1). La selección de dos puntos siempre es una recta, de tres puntos, dos puntos y una recta o dos rectas será un plano, cualquier otra selección para crear elemento es incorrecta.

Una vez creado un elemento se dibujarán todos los elementos creados hasta ese momento sin datos auxiliares y con los nombres de todos los elementos.

Puntos

Los puntos se crean a partir de sus coordenadas, estas se introducirán en el campo de texto (1), teniendo en cuenta que la coma decimales el punto '.' y el signo menos es el guion '-', todos los demás caracteres serán números. Se deben introducir tres coordenadas numéricas, en caso contrario aparecerá un mensaje de aviso. Una vez indicadas las coordenadas se presiona el botón Nuevo (6).

Se interpretan como puntos todos los datos introducidos en el campo de texto que no empiecen por 'r'/'R' o 'a'/'A'.

Rectas

Las rectas se crean a partir de dos puntos ya creados, para indicar los dos puntos hay dos opciones:

- Por el campo de texto (1): Para introducir rectas por un campo de texto se deberá inicializar con una 'r'/'R' seguida de los nombres de los puntos, por ejemplo, para una recta formada por los puntos P1 y P2 valdría con escribir 'r P1 P2'.
- Selección: Dos puntos del Listado de Puntos (3).

Una vez indicados los puntos que forman la recta se presiona el botón Nuevo (6).

Planos

Los planos se crean a partir de elementos ya creados ya sea:

- Tres puntos no alineados.
- Una recta y un punto no perteneciente a ella.
- Dos rectas paralelas o que se corten.

Para indicar los elementos hay dos opciones:

- Por el campo de texto (1): Para introducir planos por un campo de texto se deberá inicializar con una 'a'/'A' seguida de los nombres de los elementos que lo formen sin importar el orden, por ejemplo, para una recta formada por el punto P1 y la recta r1 valdría con escribir 'a r1 P1'.
- Selección: Seleccionando los elementos de los listados correspondientes, de puntos (3) o de rectas (4).

Una vez indicados los elementos que forman la recta se presiona el botón Nuevo (6).

En caso de que los elementos no formen un plano se mostrará un aviso que informe del problema.

- *Dibujar*

Se pueden hacer dibujos de todos los elementos, en cuyo caso no se tendrá en cuenta que todas las trazas de las rectas entren en las escalas, para ello, sin ninguna selección en los listados (3), (4) y (5) se presiona el botón Dibujar (7). Si hay alguna selección al presionar el

botón, sólo se dibujarán los elementos seleccionados, en este caso siempre se muestran las trazas de las rectas.

Para dibujar se tienen varias posibilidades:

Datos auxiliares:

Como se indico en el capítulo de Nomenclatura utilizada y Elementos, se permite el dibujo de los elementos con más o menos cantidad de información: Para que se muestre toda la información posible, cuando se presione el botón Dibujar (7) debe estar marcado la selección auxiliar (2), si no lo está se dibuja la información mínima.

Selección de textos:

Una vez se tienen elementos dibujados, se permite eliminar o volver a poner los nombres que identifican algunos o todos los componentes:

1. Cuando se hace doble click en un elemento creado de cualquiera de los listados (3), (4) y (5), si coincide con alguno de los dibujados, actuará sobre los nombres identificativos de este, de tal forma que si tiene los nombres identificativos los quitará y si no los tiene los colocará.
2. Al presionar el botón textos (13) si se tienen los textos colocados, se quitarán, y si no están colocados, se colocarán.

Maximizar Dibujo

Se puede maximizar el dibujo en dos dimensiones presionando la Lupa 2D (10), que abrirá una ventana con el mismo dibujo que el realizado en la ventana principal en dos dimensiones, esta ventana permite el cambio de tamaño, pudiendo adaptar la imagen a las necesidades del usuario.

Lo mismo se puede hacer para el dibujo en tres dimensiones presionando la Lupa 3D, que abrirá una ventana con el dibujo en tres dimensiones, que además permite su giro.

- ***Borrar***

Para borrar los elementos se seleccionan los elementos que se quieren eliminar y se presiona el botón Borrar (7).

Se podrán borrar varios elementos a la vez, siempre que sean del mismo tipo, con esto se pretende evitar borrar datos de una posible selección involuntaria.

No se permiten borrar elementos que están siendo utilizados para formar otros elementos, por ejemplo, si tenemos una recta ' $r_1(P_1, P_2)$ ' no se permiten borrar los puntos P_1 ni P_2 .

Con el borrado de elementos pueden renombrarse otros elementos del mismo tipo para mantener un orden lógico en el nombrado.

Una vez borrados los elementos se dibujarán todos los elementos que se mantengan sin datos auxiliares y con los nombres de todos los elementos.

- *Limpiar*

Con el botón Limpiar (9), se eliminan todos los elemento creados y se vuelve al estado inicial de trabajo, es decir, se reinicia el espacio de trabajo

- *Ayuda*

Con el botón Ayuda (12) se podrá acceder en el momento de trabajo a un breve texto que explica las posibilidades de la aplicación.